

# EXHIBIT 18

11/04/02  
J1135 U.S. PTO

00423696-1111007

11-05-02

PTO/SB/16 (10/01)

Approved for use through 10/31/2002 OMB 0651-0032

U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number

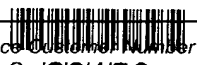
**PROVISIONAL APPLICATION FOR PATENT COVER SHEET**

This is a request for filing a PROVISIONAL APPLICATION FOR PATENT under 37 CFR 1.53(c).

Express Mail Label No.

EV188390615

C715 U.S. PTO  
60/223696  
11/04/02

INVENTOR(S)					
Given Name (first and middle [if any])		Family Name or Surname		Residence (City and either State or Foreign Country)	
Bobby		Jose		Veradale, Washington	
James		Brennan		Liberty Lake, Washington	
Edward		Casas		Vancouver BC, Canada	
<input type="checkbox"/> Additional inventors are being named on the _____ separately numbered sheets attached hereto					
TITLE OF THE INVENTION (500 characters max)					
Multi-Mac Control Techniques					
Direct all correspondence to. <b>CORRESPONDENCE ADDRESS</b>					
<input checked="" type="checkbox"/> Customer Number <b>29150</b>		 Place Customer Number Bar Code here <b>29150</b> PATENT & TRADEMARK OFFICE			
OR Type Customer Number here					
<input type="checkbox"/> Firm or Individual Name					
Address					
Address					
City		State		ZIP	
Country		Telephone		Fax	
ENCLOSED APPLICATION PARTS (check all that apply)					
<input checked="" type="checkbox"/> Specification Number of Pages <b>153</b>		<input type="checkbox"/> CD(s), Number			
<input type="checkbox"/> Drawing(s) Number of Sheets		<input checked="" type="checkbox"/> Other (specify) <b>Return Post Card</b>			
<input type="checkbox"/> Application Data Sheet. See 37 CFR 1.76					
METHOD OF PAYMENT OF FILING FEES FOR THIS PROVISIONAL APPLICATION FOR PATENT					
<input type="checkbox"/> Applicant claims small entity status. See 37 CFR 1.27		FILING FEE AMOUNT (\$)			
<input type="checkbox"/> A check or money order is enclosed to cover the filing fees					
<input checked="" type="checkbox"/> The Commissioner is hereby authorized to charge filing fees or credit any overpayment to Deposit Account Number: <b>12-0769</b>		<b>\$80.00</b>			
<input type="checkbox"/> Payment by credit card. Form PTO-2038 is attached.					
The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.					
<input checked="" type="checkbox"/> No					
<input type="checkbox"/> Yes, the name of the U.S. Government agency and the Government contract number are _____					

Respectfully submitted,

SIGNATURE

Date **11/04/02**

TYPED or PRINTED NAME

**Thomas A. Jolly**REGISTRATION NO.  
(if appropriate)**39,241**

Docket Number:

**MN1-011USP1**TELEPHONE **(509) 324-9256****USE ONLY FOR FILING A PROVISIONAL APPLICATION FOR PATENT**

This collection of information is required by 37 CFR 1.51. The information is used by the public to file (and by the PTO to process) a provisional application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 8 hours to complete, including gathering, preparing, and submitting the complete provisional application to the PTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, Washington, D.C. 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Box Provisional Application, Assistant Commissioner for Patents, Washington, D.C. 20231.

PTO/SB/17 (11-01)

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

*Patent fees are subject to annual revision*

Application Number	
Filing Date	
First Named Inventor	Jose
Examiner Name	
Group Art Unit	
Attorney Docket No.	MN1-011USP1

**Burden Hour Statement:** This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, Washington, DC 20231. **DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS.** SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

EV188390615

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
PROVISIONAL APPLICATION FOR LETTERS PATENT

**Multi-Mac Control Techniques**

Inventor(s):  
**Bobby Jose**  
**James Brennan**  
**Edward Casas**

ATTORNEY'S DOCKET NO. MN1-011USP1

**TECHNICAL FIELD**

This invention relates to wireless communications and more particularly to methods and apparatuses for use in wireless data packet communications systems capable of supporting multiple point-to-point links, packet-by-packet steering, and the like.

**BACKGROUND**

Conventional wireless local area network (LAN) systems typically employ a micro-cellular arrangement, wherein, for example, a small base station, often referred to as an Access Point (AP), is configured to communicate with wireless devices attached to computing devices, such as, laptops or other portable data appliances. These APs have a limited range, typically 20 to 200 feet for an IEEE 802.11(b) system. Thus, to cover a large area a system may require a plurality of APs. This can be costly and tends to complicate the wireless system.

There is a need for improved methods and apparatuses that can provide wireless communications.

**DESCRIPTION**

The following description sets forth a specific embodiment of a wireless communications system that incorporates elements recited in the appended exemplary claims and others. The embodiments are described herein and in the attached documentation. However, the description itself is not intended to limit the scope of this patent. Rather, the inventors have contemplated that the invention might also be embodied in other ways, to include different elements or

1 combinations of elements similar to the ones described in this document, in  
2 conjunction with other present or future technologies.

3 With this in mind, methods, apparatuses, and systems are provided for a  
4 project code-named "Little Joe" developed by Vivato Incorporated (formally  
5 known as Mabuhay Networks) having research and development offices in  
6 Spokane, WA, and headquarters in San Francisco, CA.

7 In accordance with certain exemplary aspects of the present invention,  
8 Multi MAC Control for a coordinating 802.11 MACs is provided for use with  
9 Smart Antenna Systems. The Multi Mac Control (MMC) can configured to avoid  
10 unnecessary collisions of downlink traffic with ongoing uplink traffic in a  
11 system/product with multiple independent 802.11 MACs. In this exemplary case  
12 the multiple MACs use the standard 802.11 protocol to share access to the  
13 channel. MMC provides the ability to control multiple independent off the shelf  
14 MACs adaptively using an external controller as in the Vivato Little Joe product.

15 In accordance with certain exemplary aspects of the present invention, the  
16 Little Joe architecture has one WLAN AP card per beam. This allows each card to  
17 operate with its own independent MAC controller. All of the APs listen  
18 simultaneously but only one of them can transmit at a time. Each card's MAC  
19 protocol enforces the "one transmitter at a time" rule through the use of it's CCA  
20 circuitry. Any transmission from any AP on any beam will interfere with and  
21 cause the loss of any frames currently being received on other beams. However,  
22 since the APs on different beams can hear each other's transmissions, there should  
23 be no collisions after the initial downlink DATA or CTS frame (as a result of other  
24 beams setting their NAV timers). No special MAC software is required at the  
25

1 client. Each beam appears as a different AP and a client associates with whichever  
2 beam it thinks is providing best coverage.

3 Incorporated herein are the following appendices:

- 4 A. Document: Little Joe Functional Specification (113 Pages)  
5 B. Document: Multi-MAC Controller Design (13 Pages)  
6 C. Presentation Slides: Little Joe Multi-MAC Controller (11 Pages)  
7 D. Presentation Slides: Little Joe MAC Problems (11 Pages)
- 8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25

**EXEMPLARY CLAIMS**

1. A method comprising:  
using a Multi-MAC Control (MMC) for a coordinating MACs in a  
communications systems having at least one phase array antenna.

2. The method as recited in Claim 1, wherein the MMC is configured to  
significantly avoid collisions of downlink traffic with ongoing uplink traffic in  
said communications system.

3. The method as recited in Claim 1, wherein said MACs include a  
plurality of independent 802.11 MACs.

4. A Multi-MAC controller.

5. A communications system comprising at least one Multi-MAC  
controller.

6. A Little Joe device.

7. A communications system having at least one Little Joe device.

8. A propagated signal from a Little Joe device.



6404235905 11109002

EV188390615

A

## Little Joe Functional Specification

[113 pages]

A-1



Case 2:23-cv-00202-JRG-RSP Document 86-5 Filed 08/12/24 Page 10 of 156 PageID #: 2444

Little Joe Functional Specification

Document History

Contributor(s)	Description	Date

A-2

## Table of Contents

Contents.....	3
Part 1: Radio and MAC.....	7
1. Introduction.....	7
2. System Architecture .....	8
2.1. System.....	8
2.2. Deployment Scenarios .....	10
2.2.2. Typical Building Structure .....	12
2.2.3. Traffic Load Assumptions .....	12
2.3. Interference effects .....	13
2.4. Coverage requirements .....	13
2.5. Throughput and delay requirements .....	15
2.5.1. Performance Metrics .....	15
2.6. Performance limiting issues .....	15
2.7. Technology.....	16
2.8. System Block Diagram .....	16
3. The Beamforming Network .....	19
3.1. The Butler Matrix .....	19
3.2. Ideal Beam Patterns .....	20
3.3. Receive Windowing.....	23
3.3.1. The Hamming Window.....	23
3.4. Complementary Transmit Beamforming .....	26
4. Link Budgets .....	28
4.1. Link Power Budget.....	28
4.2. Link Budget Parameters.....	30
4.2.1. Panel Transmit Power: FCC EIRP Power Limits.....	30
4.3. Characteristics of Conventional 802.11b Equipment .....	31
4.3.1. Panel Antenna Gain .....	33
4.3.2. Gain Reduction due to Scattering .....	33
4.3.3. Client Antenna Gain.....	34
4.3.4. Client and Panel Noise Figures.....	34
4.3.5. Effect of Shadow Fading .....	35
4.3.6. Effect of Rayleigh Fading .....	36
4.4. Path Loss Models .....	36
4.4.1. Propagation in Free Space (LOS).....	37
4.4.2. Propagation by Diffraction (NLOS) .....	37
4.4.3. Propagation by Transmission (OBS) .....	37
4.4.4. Outdoor-Indoor Path Loss .....	38
4.4.5. Indoor-Indoor Path Loss.....	39
4.5. Link Budget Results .....	39
5. Transmit Power Control .....	42
5.1. Power Control Requirements .....	42
5.2. Power Control State Transition Diagram.....	43
5.3. Power Control SNR Measurement .....	44

A-3

## Little Joe Functional Specification

6.	Media Access Control (MAC).....	44
6.1	ViVATO Panel MAC Operation.....	44
6.1.1.	MAC Modes of Operation.....	45
7.	Multi-Radio Transmit Control.....	46
7.1.	Multi-MAC Control (MMC).....	47
7.2.	MMC Overview.....	47
7.3.	MMC Provisioned Parameter.....	48
8.	Intra-Panel Roaming.....	48
8.1.	Roaming Requirements.....	49
8.2.	Beam-Switching Algorithm.....	49
8.2.1.	Beam-Switching State Transition Diagram.....	50
8.3.	IAPP (Seamless Roaming).....	52
9.	Channel Assignment.....	52
9.1.	Channel Assignment Provisioned Parameters.....	53
9.2.	Channel Assignment Internal Parameters.....	53
9.3.	Channel Assignment Metrics.....	53
9.4.	Channel Assignment Algorithm.....	55
9.4.1.	Channel Assignment Preprocessing.....	55
9.4.2.	Block-based Channel Assignment Algorithm.....	56
9.4.3.	Emergency exit.....	60
10.	Downlink Traffic-Shaping.....	61
10.1.	Traffic-Shaping Requirements.....	61
10.2.	Traffic-Shaping Architecture.....	61
10.3.	Traffic-Shaping Functional Description.....	63
10.3.1.	Leaky Bucket Algorithm.....	63
10.3.2.	Granularity of Operation.....	64
10.3.3.	Dynamic Parameter Update.....	64
10.3.4.	Operating Point Estimation Algorithm.....	65
10.4.	Provisioned and Internal Parameters.....	66
11.	The Scanning Radio.....	67
11.1.	Scan Mode.....	67
11.2.	Roaming.....	68
11.3.	State Transitions.....	68
12.	References.....	69
Part 2:	Software System Architecture.....	72
13.	Introduction.....	72
14.	Software Overview.....	74
14.1.	Software not covered by this Document.....	74
14.2.	Software Module Overview.....	75
15.	Control Plane.....	76
16.	Drivers.....	76
16.1.	Console Driver.....	76
16.2.	Ethernet Drivers.....	77
16.2.1.	Secure Management.....	77
16.2.2.	Backhaul / Daisy-chain for the Next Panel.....	78

A-4

16.3	Source for Wireless Drivers .....	79
16.4	Searcher and Merlin Interfaces and Functions .....	79
16.5	Scheduler / Shaper .....	81
16.6	RRM (Radio Resource Management) .....	82
16.7	FLASH .....	83
This table shows initial estimations of management module resource usage. Need clarification of data items in the table. Some of these items are daemons, running all of the time. Others are one instance per invocation. Cish is dependent on lots of other elements, such as ipchains, brctl and ifconfig .....		
17.	Management Interfaces .....	83
17.1	HTTP .....	85
17.2	CISH (CLI) .....	86
17.3	User Manager .....	87
17.4	SSHD .....	87
17.5	Telnet .....	87
17.6	SNMPD .....	87
18.	Environmental Control .....	87
18.1	Temperature and Fan Control .....	88
19.	Wireless Control .....	88
19.1	Wireless Bridging .....	88
19.2	Centralized Bridging .....	89
19.4	Radio Configuration Manager .....	90
19.4	Inter-card Roaming Manager .....	90
Mabuhay Enhanced Performance System (MEPS) .....		90
Security .....		90
19.4	Authentication Manager .....	90
19.4	Authentication .....	90
19.4	802.1x / EAP Authentication Mechanism .....	91
19.	Other Control .....	91
19.4	DHCP Client .....	91
19.	DHCP Server and Network Address Translation (NAT) .....	92
19.	Typical Packet Walk (Bridged) .....	93
19.	Typical Packet Walk (NAT and DHCP Server) .....	93
19.	Diagnostics .....	95
19.	Out-of-the-Box Power-up Sequence .....	95
19.	Utilities .....	95
19.4	FTP Client (for downloading new Images) .....	95
19.	FTP Server (for Serving images to other panels) .....	95
19.	Data Packet Interfaces .....	96
19.	Appendix A – PCI / PCI Bridge Support in Linux .....	97
19.4	Example PCI Based System .....	97
19.4	PCI Address Spaces .....	97
19.4	PCI Configuration Headers .....	98
19.4	Layout of the 256 byte PCI configuration header .....	99
19	Vendor Identification .....	99

A-5

CONFIDENTIAL

# Little Joe Functional Specification

19.	Device Identification.....	99
19.	Status.....	99
19.	Command .....	99
19.	Class Code.....	99
19.4	Base Address Registers.....	100
19.4	Interrupt Pin.....	100
19.4	Interrupt Line.....	100
19.	PCI I/O and PCI Memory Addresses.....	100
19.	PCI-ISA Bridges.....	100
19.	PCI-PCI Bridges.....	101
19.	PCI-PCI Bridges: PCI I/O and PCI Memory Windows .....	101
19.	PCI-PCI Bridges - PCI Configuration Cycles and PCI Bus Numbering.....	101
	Type 0 PCI Configuration Cycle Figure: .....	101
	Type 1 PCI Configuration Cycle Figure: .....	102
19.	Linux PCI Initialization .....	103
19.4	The Linux Kernel PCI Data Structures.....	103
19.4	The PCI Device Driver .....	104
19.	Configuring PCI-PCI Bridges - Assigning PCI Bus Numbers .....	105
b.	PCI I/O and PCI Memory Windows .....	106
a.	PCI-PCI Bridge Numbering: Step 1 .....	106
b.	PCI-PCI Bridge Numbering: Step 2.....	106
d.	PCI-PCI Bridge Numbering: Step 4.....	107
e.	PCI BIOS Functions .....	108
f.	PCI Fixup .....	108
g.	Finding Out How Much PCI I/O and PCI Memory Space a Device Needs .....	108
a.	PCI Configuration Header: Base Address Registers .....	109
b.	Allocating PCI I/O and PCI Memory to PCI-PCI Bridges and Devices .....	109

A-6

## Part 1: Radio and MAC

### Overview

This document describes the LittleJoe 802.11b WiFi switch (WS) from a functional point-of-view. The document includes system description, the beamforming functions, radio resource management (RRM), and the media access control (MAC) functionality. Higher layer functionality, network management and other functions are described in separate documents M. Brewer, D. Lohman, et. al. "Software System Architecture Document", V1.

The functional descriptions in this document provide details of the functions performed by any subsystem. The final architectural design is discussed in separate documents including the control and interface functions between the subsystems.

The product concept is discussed in Introduction.

The high level system architecture is presented in

### System Architecture

. The beamforming network which is the core technology of this product is described in The Beamforming Network.

The radio link budgets are presented in Link Budgets

The transmit power control scheme is described in Transmit Power Control. The media access control (MAC) is described in Media Access Control (MAC). The multi-radio transmit control, multi-MAC Control (MMC) and roaming are discussed in

### Multi-Radio Transmit Control

Multi-MAC Control, and Intra-Panel Roaming respectively. The channel assignment and traffic shaping algorithms are described in Channel

A-7

## Little Joe Functional Specification

Assignment and Dow respectively. The scanning radio functions are described in

### The Scanning Radio

## 1. Introduction

LittleJoe is ViVATO's first-generation long-range packet switch built according to the 802.11 standard. It seamlessly supports 802.11b clients. The LittleJoe WiFi switch features:

- linear array of 16 antennas providing up to 29 dBi of gain
- Butler-matrix beamforming
- Complementary beamforming
- Multi-MAC controller
- Multi-channel operation
- a high sensitivity RF front end
- Agere 802.11b MAC and baseband processor chips
- Custom logic and software for integration
- Security enhancements

There are two configurations:

- DirectedPacket™ 1 (DP2310) transmits on one channel at a time
- DirectedPacket™ 3 (DP2330) transmits up to 3 channels at a time

These can be fit into two types of antennas/enclosures:

- Indoor half height: 1m wide by 0.5 m high
- Outdoor full height: 1m wide by 1m high

## 2. System Architecture

In this section we review the ViVATO Packet Switch concept and it's advantages compared to existing APs in the market. We also describe ViVATO's high-level system block diagram. The details of the subsystems are then described in different sections within the document.

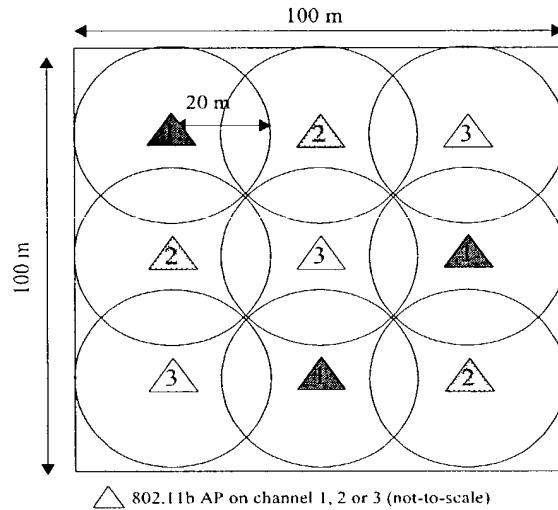


### 17.3 System Concept

Figure 2 A typical deployment for conventional 802.11 APs is shown in *Reference deployment for 802.11 networks*. The example deployment scenario assumes a 10,000 m<sup>2</sup> coverage area covered by 9 conventional APs each with a cellular coverage radius of 20 m. The assumed coverage cells are circular each with a radius of 20 m. The reference LittleJoe deployment is shown in *Reference LittleJoe indoor deployment*.

One LittleJoe panel (with a range of about 140 m) is used (typically at the corner of a building indoors or mounted on a tall structure outdoors) to provide service to the whole coverage area. In some deployment scenarios, there are regions where LittleJoe cannot provide service due to shadowing or severe scattering. This is represented by the gray shaded circle in *Reference LittleJoe indoor deployment*.

Such areas will be covered using ViVATO APs. In the above example, 9 regular APs have been replaced by one LittleJoe Packet Switch and one ViVATO AP. This reduces the cost of network deployment. In addition, it is possible to provide outdoor coverage by deploying the unit outdoors.

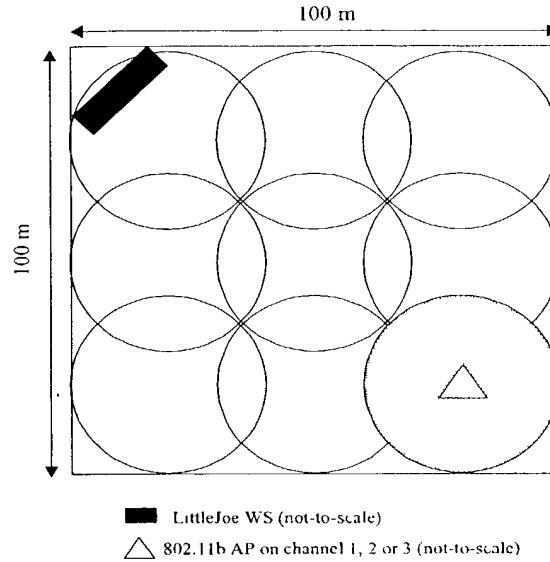


A-9

US 10,114,222 B2

## Little Joe Functional Specification

Figure 1 Reference deployment for 802.11



networks

Figure 2 Reference LittleJoe indoor deployment

### 17.3 Deployment Scenarios

LittleJoe is a long-range WiFi switch and therefore it can support many different applications. Nevertheless, there are five reference deployment scenarios identified.

- Indoor Office
- Indoor Warehouse
- Outdoor Campus
- Outdoor Hotel
- Outdoor ISP

#### 17.1 Typical Mounting Conditions

There are two types of LittleJoe units: half height indoor and full height outdoor.

A-10

### 2...1. Indoor Deployment

A LittleJoe WS is placed inside the building to provide coverage throughout the whole building.

Typically a half height LittleJoe panel is installed on a corner wall inside an office building or a warehouse. The ceiling height is typically between 3 to 4m for the office building and 4 to 8m in a warehouse.

The center of the panel is typically installed at 0.25m below the ceiling. The unit should be mounted away from nearby scatterers or obstructions.

### 2...2. Outdoor Deployment

Typically one to four co-located full height LittleJoe panels are installed in a desirable location on campus grounds, each panel providing 100° coverage. The typical antenna height is 4 to 20m. The furthest building to LittleJoe installation site is no more than 200m away. The buildings are low-rise (6 stories or less). The scatterers are mostly local to the client and not close to the LittleJoe panel.

For the Outdoor Hotel model, the typical antenna heights would be about 4 - 8m and the building would be no more than 20m away from the panel.

The typical ranges and mounting conditions for the different deployment scenarios are summarized in *The range and antenna dimensions and heights for different de.*

	indoor office	indoor warehouse	outdoor office	outdoor Hotel	outdoor ISP
range	< 150m	200m	300m	100m	2 km
antenna size	1m x 0.5m	either	1m x 1m	1m x 1m	1m x 1m
antenna height	3 to 4m	4 to 8m	4 to 20m	4m to 8m	10 to 50m

A-11

## Little Joe Functional Specification

Table 1 *The range and antenna dimensions and heights for different deployment scenarios*

## 1.1.1

## 17.1 Typical Building Structure

The exterior walls are typically concrete or other hard construction material. There are usually many windows and metal on the exterior of the building.

The interior walls are not hard structures. They are typically made of drywall with wood or metal studs. The floors are typically made of concrete. The interior of the building has mostly open offices with soft enclosed cubicles. There may be some dry-wall enclosed offices and conference rooms with or without interior windows.

The unit is built with sufficient link budget to operate with the assumptions above. However, a site may have areas of exceptionally high path loss. These poor coverage areas may be serviced using ViVATO APs<sup>1</sup> connected to the LittleJoe panel through the backbone network or in-band 802.11 signalling.

## 1.1.2

## 17.1 Traffic Load Assumptions

There may be more than 200 associated users inside the building. Most users are connected to a wired network and hence do not generate any traffic unless when they require portable connectivity. Nevertheless, they may send probe request frames regularly. The network is expected to perform well with a maximum of 50 active users each with a profile of a "typical" LAN user. The traffic load assumptions for the different deployment scenarios are summarized in *Traffic load assumptions for different deployment scenarios*.

	indoor office	indoor warehouse	outdoor office	outdoor Hotel	outdoor ISP
associated users	> 200	> 200	>200	>200	400

<sup>1</sup>A ViVATO AP (also known as the Pollen8) is an open-source AP whose software has been updated to work effectively with the LittleJoe WPS. The details will be described in separate documents.

A-12

<b>active users</b>	20	50	40	10	40
<b>profile</b>	office LAN	short transactions	office LAN	home LAN	home LAN

Table 2 *Traffic load assumptions for different deployment scenarios.*

### 17.3 Interference effects

The following are the major sources of interference to the ViVATO network:

- Microwave ovens: create a coverage hole of 5-10 m while they are transmitting. It is recommended that the panel be placed as far away as possible from microwave ovens. It is also advisable to shield the microwave ovens to reduce interference to the network.
- Cordless phones: many operate on the same band and can hence interfere severely with both the panel and client transmissions. It is advised that cordless phones in office and warehouse deployments not be used. For the ISP deployment scenario, the cordless phones will be a significant source of interference and may reduce the networks performance to unacceptable levels.
- 802.11 private LANs: the ViVATO network shares the frequency resources with other 802.11 networks.

### 17.3 Coverage requirements

Coverage is defined as a packet error rate of 10% or better at a specified data rate. LittleJoe should have an indoor coverage of 85% coverage at 11 Mbits/s and 95% at 5.5 Mbits/sec. ViVATO APs are used to fill in large coverage holes in rare places (no more than 10% of the coverage area) due to severe deployment conditions.

The nominal expected coverage for the different deployment scenarios are summarized in *Coverage for different deployment scenarios..*

	<b>indoor office</b>	<b>indoor warehouse</b>	<b>outdoor office</b>	<b>outdoor Hotel</b>	<b>outdoor ISP</b>
<b>range</b>	150 m	200 m	300 m	100 m	2 km
<b>11 Mb/s coverage</b>	85%	85%	85%	85%	75%

A-13

Exhibit A-1: Little Joe Functional Specification

Little Joe Functional Specification

5.5 Mb/s coverage	95%	95%	95%	95%	80%
coverage holes	5-15%	5-15%	5-15%	5-15%	5-25%

A-A

Table 3 Coverage for different deployment scenarios

## 1.2

### 17.3 Throughput and delay requirements

The user experience should be similar to that of a deployment of multiple low-range access points shown in *Reference deployment for 802.11 networks*.

#### 17.1 Performance Metrics

The following performance metrics measure the quality of service provided by LittleJoe compared to conventional 802.11 deployments. These are non-real-time traffic measures G. Anastasi, et. al., "MAC Protocols for Wideband Wireless Local Access: Ev:

- Packet queuing delay: the time elapsed from the time a packet is generated until it is ready to contend for access to the channel
- Packet MAC delay: the time elapsed between the time a packet is ready to contend until the beginning of its successful transmission
- Packet access delay: the sum of the queueing delay and MAC delay
- Packet transmission time: the time between the start of a transmission to its successful completion.
- Packet throughput: the number of bits transmitted in a packet divided by packet transmission time.
- Packet loss rate: the number of packets whose transmission time exceeds TCP time-out.

### 17.3 Performance limiting issues

The following effects are the most detrimental to LittleJoe performance:

- Nearby large scatterers
- hard structures (hard walls, metal objects, etc.)
- competing 802.11 networks
- high angle spread environment
- microwave ovens
- 2.4 GHz cordless phones

A-15

## Little Joe Functional Specification

### 1.3

#### 17.3 Technology

The technologies enabling the increased coverage for LittleJoe are:

- Phased Arrays
  - vertical antenna gain
  - directional receive and transmit patterns using a Butler matrix
  - taking advantage of point-to-point power limits
- Technologies to enable directional antennas with 802.11
  - complementary beamforming
  - Multi-Radio Transmit Control
  - Multi-MAC control

Additional capacity gain is obtained using:

- multi-channel operation
- seamless roaming
- traffic-shaping

### 1.4

#### 17.3 System Block Diagram

This section includes the functional block diagram of LittleJoe. The single channel product (DP2310) can operate on a single channel at any time. The 3 channel product call DP2330 can operate on three different channels simultaneously.

Figure 3 The block diagram for The DP2330 product is shown in *The DP2330 high-level block diagram*.

. There are 16 antenna elements each with its own RF front-end. The receive port of each circulator is followed by an LNA before the Butler matrix. The circulators ensure that:

- signals arriving at the antenna elements are directed to the receive chain and isolated from the transmit chain
- signals transmitted through the transmit chain are directed to the antenna elements and isolated from the receive chain.

A-16



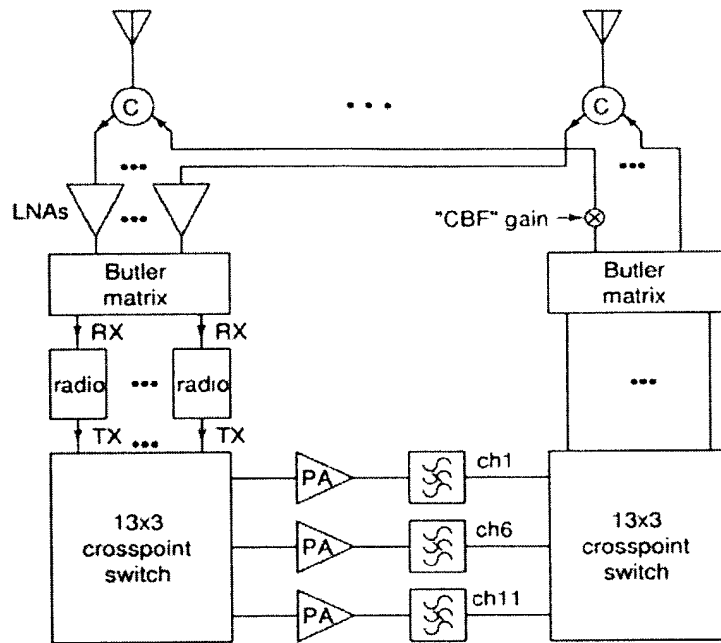


Figure 3 The DP2330 high-level block diagram

Each PA is followed by a ceramic bandpass filter. To allow for simultaneous transmit and receive on adjacent channels the LNA can handle strong input signal with no distortion and the post-PA bandpass filters have high adjacent-channel rejection.

The PA output powers can be reduced from their maximum levels in by least 24 dB (4 steps of 6 dB).

A single-channel cost-reduced model (DP2310) may use switches instead of circulators and eliminate the crosspoint switches and filters.

## 1.5

The Butler Matrix described in detail in

The Butler Matrix

A-17

#### Little Joe Functional Specification

is a network of passive hybrid power dividers and fixed phase shifters with 16 inputs and 16 outputs. The Butler matrix produces 16 orthogonal beams each pointing in a different direction. However, since the pattern is distorted at the extreme angles, only 13 ports of the Butler matrix on the radios side are used.

Figure 3 Each Butler matrix port at the radio side is connected to a splitter which splits the signal to a WLAN radio and a 13-way switch which is connected to a scanning radio (not shown in *The DP2330 high-level block diagram*)

Figure 3 ).

#### **1.6 Each WLAN radio is built using the Agere 802.11b chipset<sup>2</sup>. The Agere MAC controller is used and runs AP firmware. See**

ViV for more details.

The receive ports of the 13 radios are connected to 13 ports of the receive Butler matrix and the transmit ports are connected to a 13x3 switch that can connect each radio to any of the 3 PAs. The PA outputs are connected to a second 3x13 switch that connects each of the 3 PAs to any of the transmit Butler matrix ports.

The beam patterns and the numbering convention at the Butler Matrix ports with and without windowing are described in Bea.

Figure 4 A scanning receiver in "promiscuous mode" and a "beam" antenna switch are used to obtain signal strength and interference information from stations on different beams and channels as shown in *The scanning radio*.

---

<sup>2</sup>The choice of the chipset vendor is subject to change

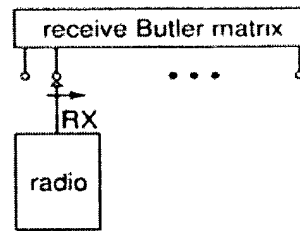


Figure 4 *The scanning radio.*

### 3. The Beamforming Network

The increase in the range of the product is obtained by beamforming. The beamforming technology is based on multiple simultaneous reception of signals from different directions through a Butler matrix and transmitting a directional signal through a similar entity.

Additionally, receive windowing is applied to the Butler matrix to help mitigate the near-far problem, and complementary beamforming is applied to reduce the effect of “hidden beam”. The functional details of the beamforming network are discussed in this section.

#### 1.7

#### 17.3 *The Butler Matrix*

The Butler matrix is a network of passive hybrid power dividers and fixed phase shifters with 16 inputs and 16 outputs. The Butler matrix produces 16 orthogonal beams each pointing in a different direction. The Butler matrix is a theoretically loss-less network that provides maximal gain from the antenna aperture. *Beam and DFT numbering of the Butler matrix ports* ( shows the beam and DFT numbering of Butler Matrix’s input and output ports. The beams are numbered 0 to 15 from left to

A-19

## Little Joe Functional Specification

right.

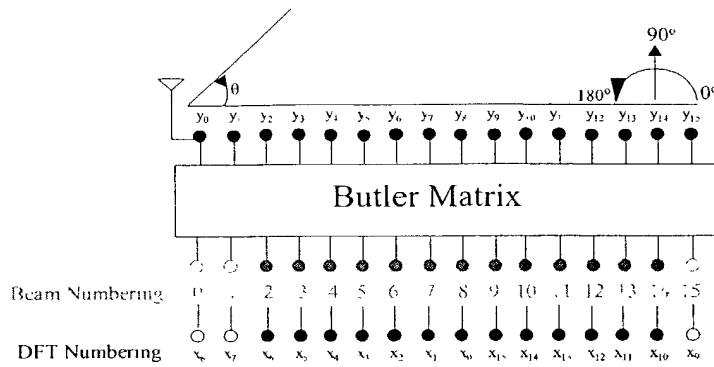


Figure 5 Beam and DFT numbering of the Butler matrix ports (top view).

The mathematical description of the Butler matrix response is the same as a Discrete Fourier Transform (DFT).

If the signals at the input of the Butler matrix (radio side) are denoted by  $x_n$  where  $n=0$  to 15. The output of the Butler matrix (antenna side) is

$$y_k = \frac{1}{N} \sum_{n=0}^{N-1} x_n e^{-j \frac{2\pi nk}{N}} \quad k = 0 \quad N-1$$

then:

where  $N=16$ . The amplitude response (far-field transmit beam pattern) of the Butler matrix

$$z(\theta) = \sum_{k=0}^{N-1} y_k e^{j k \pi \cos(\theta)}$$

is:

The far field pattern for a signal at radio port  $l$  ( $x_l$ ) is

$$z_l(\theta) = \sum_{k=0}^{N-1} x_l e^{-j \frac{2\pi k l}{N}} e^{j k \pi \cos(\theta)}$$

then:

As shown in Ed Casas, LittleJoe Beamforming, , for a unity input signal, this may be described by the following

A-20

formula

$$|z_l(\theta)| = \frac{1}{N} \sqrt{\frac{1 - \cos(\phi)}{1 - \cos(\frac{\phi}{N})}} \quad k = 0 \quad N-1 \quad \phi = N\pi \cos(\theta) - 2\pi l$$

Equivalently, the received signal ( $x_k$ ) on port  $k$  on the radio side due to a signal incident from angle  $\theta$  may be described by the same formula.

## 1.8

### 17.3 Ideal Beam Patterns

The ideal beam patterns of shows the beam patterns ( $20\log(|z_k(\theta)|)$ ) of a 16 element Butler matrix as a function of the signal's angle of incidence  $\theta$ .

There are some important observations:

- the beams get wider as we move towards the ports at the outer edges (end-fire)
- the beams at the extreme edges are very wide
- the side lobes are at almost -13 dB
- The cross-beam loss is about 4 dB

Since the beams due to signals transmitted into ports  $x_7$ ,  $x_8$  and  $x_9$  are very wide, they are left unused. This means that 13 radios are used. This limits the field of view from about  $40^\circ$  to  $140^\circ$  or to about  $100^\circ$ . Therefore each LittleJoe panel has a field of view of  $100^\circ$ . In other words, to provide  $360^\circ$  coverage, 4 panels are needed.

A-21

08/12/24 13:05:05, 1.1014002

## Little Joe Functional Specification

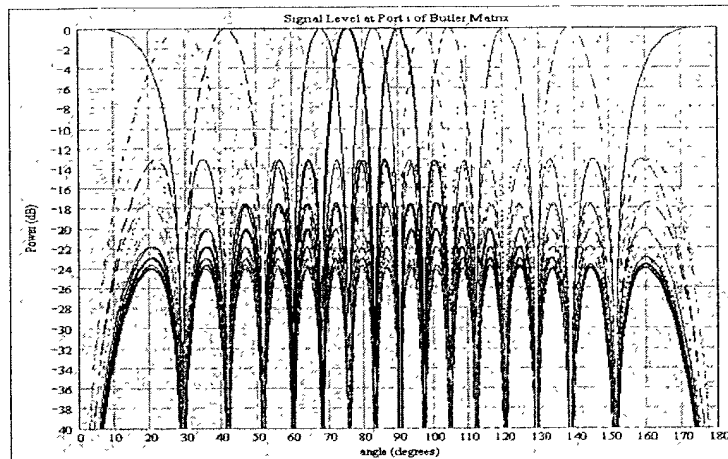


Figure 6 The ideal beam patterns of a 16 element Butler matrix.

A radio is attached to each Butler matrix port. Each radio has a boresight direction (the direction with peak signal level). The boresight angles for the Butler matrix ports are shown in *The boresight directions of Butler matrix ports*.

The port numbering in the second column of *The boresight directions of Butler matrix ports* is based on the DFT index.

To avoid confusion, we have agreed on a single beam numbering convention. The numbering scheme assigns beam numbers starting from the boresight angle of zero and increasing thereafter, as shown in column 1 of *The boresight directions of Butler matrix ports*. In this case, beams number 0, 1, and 15 are left unused and the beam at 90° boresight is beam number 8.

For the purposes of complementary beamforming, it is required to identify antenna element 0. This is the antenna port that has zero phase shift relative to all the radio ports

The relatively high sidelobes levels are undesirable for the receiver. 802.11 clients do not use transmit power control. This results in a wide range of received signal levels at the switch. In both indoor and outdoor deployment the ratio of propagation distances might easily exceed a factor of 10 (near-far effect). This results in path loss differences of more than 30 to 40 dB. Therefore, the sidelobes need to be reduced. This is accomplished by receive windowing described in the next section.

A-22

Butler matrix port		boresight angles
Beam Numbering	DFT Numbering	(degrees)
0	8	0.00, or 180.00
1	7	28.96
2	6	41.41
3	5	51.32
4	4	60.00
5	3	67.98
6	2	75.52
7	1	82.82
8	0	90.00
9	15	97.18
10	14	104.48
11	13	112.02
12	12	120.00
13	11	128.68
14	10	138.59
15	9	151.05

Table 4 The boresight directions of Butler matrix ports.

### 17.3 Receive Windowing

Tapering the illumination of the array (“windowing”) reduces the sidelobe levels. The resulting trade-offs are:

- an increase in the main lobe width
- a decrease in received SNR

Therefore, the decision to use windowing may depend on the environment. For instance, in a situation where an outdoor LittleJoe unit services clients in only one building, the received power dynamic range may be relatively small and hence higher sidelobe levels may not be

A-23

Little Joe Functional Specification

harmful. On the other hand, a decrease in received signal power may not be acceptable due to range requirements. Thus the selection of a window function will depend on the expected dynamic range of the received signals (the severity of the near-far problem).

One of two windows are selected at provisioning.

- Rectangular window (no windowing)
- Hamming window (default value).

The receive window type is configured by the network element management software. The default setting for the windows is the Hamming window described below.

### 17.1 The Hamming Window:

Figure 7 A popular window with relatively low side lobes is the Hamming window shown in *The Hamming window*.

Figure 7 The Hamming window reduces the peak sidelobe level from -13 to -41 dB but doubles the main lobe null-to-null beamwidth and increases the 3 dB beamwidth from about 6° to about 10°. The Hamming window shown in *The Hamming window*.

is described by the following formula:

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \quad n = 0 \dots 15$$

where  $N=16$  The window has a bell shape. It reduces the signal power received from the outer elements of the antenna

A-24



array.

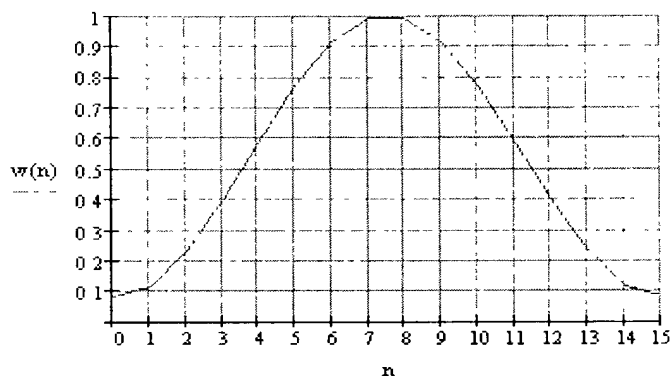


Figure 7 The Hamming window.

The window is symmetric and is implemented using a network of attenuators which attenuate the signal from each element by the appropriate Hamming window coefficient shown in *The Hamming window coefficients*.

elements	0,15	1,14	2,13	3,12	4,11	5,10	6,9	7,8
gain	0.08	0.12	0.23	0.4	0.59	0.77	0.91	1.00
gain (dB)	-21.9	-18.4	-12.7	-8.0	-4.6	-2.3	-0.8	0.0

A-25

## Little Joe Functional Specification

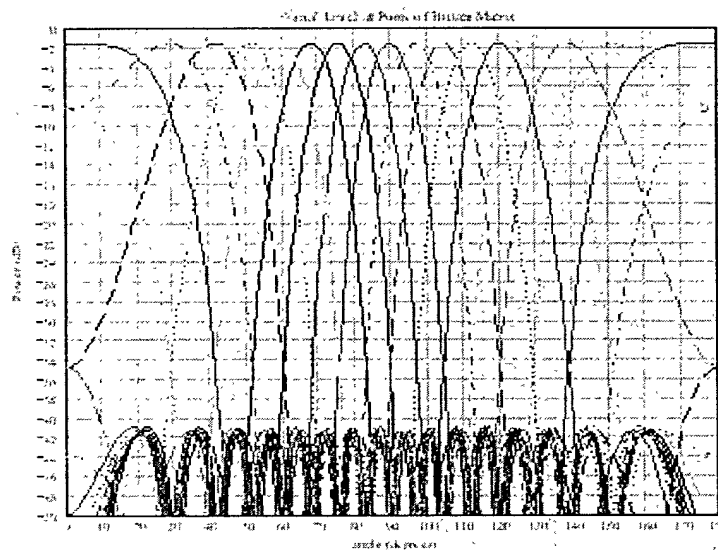
Table 5 *The Hamming window coefficients*

To normalize the effect of the window coefficients to achieve the same SNR, the receiver RF circuit should increase the voltage gain to each antenna element by the factor of:

$$G_n = \frac{N}{\sum_{n=0}^{N-1} w(n)^2} = 1.64 \quad N = 16$$

This translates to an adjusted power gain of about 4.3 dB.

Figure 8 The beams at the radio side of the Butler matrix with a Hamming window are shown in *The Butler matrix receive beam shapes with a Hamming window*



A-26

Figure 8 *The Butler matrix receive beam shapes with a Hamming window*

Figure 8 As shown in *The Butler matrix receive beam shapes with a Hamming window*

, the main beams of the Butler matrix are wider (3 dB beamwidth of about  $10^\circ$ ) but the sidelobes have been suppressed to -41 dB. There is also approximately 1.5 dB loss in SNR due to receive windowing. The Butler matrix ports 7, 8 and 9 are even worse than the non-windowed case and deemed unusable hence limiting the field of view from approximately 40 to 140 degrees.

The cross-beam loss (or the pointing loss) is about 1.5 dB.

## 1.9

### 17.3 Complementary Transmit Beamforming

As shown in *The ideal beam patterns of*, the transmit beams have very deep nulls in certain directions and the lowest sidelobe levels are around 14 dB down from the main lobe's peak.

With complementary beamforming, we intend to reduce the effect of the nulls and increase the sidelobe levels without a severe power penalty to the main beam. This is done to reduce the effect of the "hidden beam"<sup>3</sup>.

Figure 9 The complementary beam is formed by increasing the gain at the antenna element 0. This is shown in *g*

---

<sup>3</sup> The media access technique in 802.11 is Carrier Sense Multiple Access (CSMA). Forming directional transmit beams has the side effect of hiding the transmitted energy from some clients in the network; i.e., negatively impacting the carrier sense mechanism in the network. A client measures the energy transmitted from APs and other clients. If it cannot detect the presence of other transmissions, it attempts to access the medium. Therefore, when directional beams are used, many clients detect the medium as idle when in fact it is busy. This has an effect on the performance of the network. We call this phenomenon the "hidden beam" problem.

A-27

# Little Joe Functional Specification

using the DFT numbering convention. This is the only port whose output is the addition (with no phase) of all the input ports ( $y_0 = x_0 + x_1 + \dots + x_{N-1}$ ).

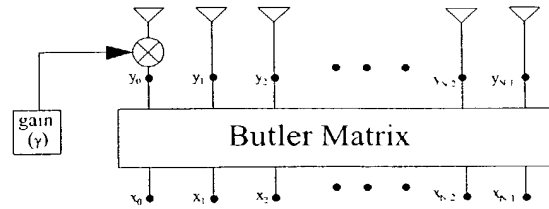


Figure 9 Block diagram of complementary beamforming.

Mathematically, this may be described as:

$$y_i = \begin{cases} \gamma y_i & i = 0 \\ y_i & \text{otherwise} \end{cases} \quad \gamma \geq 1 \quad 0 \leq i \leq N-1$$

To ensure the same output power as with no complimentary beamforming the output voltage on all the ports should be adjusted

$$G_s = \sqrt{\frac{N}{\gamma^2 + N - 1}}$$

by the scaling factor:

It may be shown that the power penalty for the main beam

$$\Delta P = \frac{(\gamma + N - 1)^2}{N(\gamma^2 + N - 1)}$$

is:

$$\Delta P_{dB} = 10 \log \left( \frac{(\gamma + N - 1)^2}{N(\gamma^2 + N - 1)} \right)$$

or in dB:

For instance, for a 16 element array, if  $\gamma=3.5$ , the power loss is about 1 dB. It is desirable to have  $\gamma$  as a parameter that may be changed at provisioning.

The Butler matrix output due to a shows the shape of the transmit beam due to a signal at port 0 of the Butler matrix with and without complementary beamforming. The output with complementary beamforming has higher sidelobes in all directions and removes all the deep nulls except for the nulls on the main beam. The main beam's peak power is about 1 dB lower

A-28

05/04/2023 06:43:05 v. 31.0004.000

than that without complementary beamforming

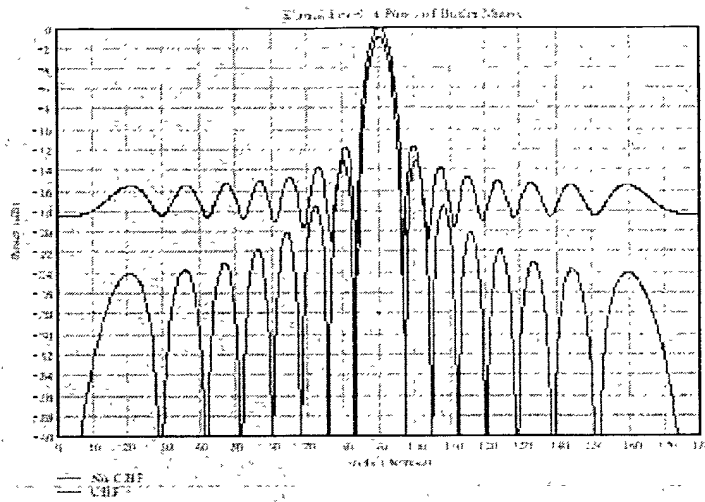


Figure 10 The Butler matrix output due to a signal at  $x_0$  (beam 8) with complementary beamforming ( $\gamma=3.5$ )

## 4. Link Budgets

The link budgets are prepared to provide a basis for making design decisions that might affect the coverage area of LittleJoe. Additionally, they provide guidance to potential customers about the improvement in coverage area they can expect from a LittleJoe panel compared to a conventional AP. In this section, we provide a number of models for indoor and outdoor applications and provide best-case, worst-case and nominal range estimates for the different deployment scenarios described in Deployment Scenarios.

Throughout this section the term panel refers to the LittleJoe packet switch and client refers to a standard IEEE 802.11 WLAN client card.

### 1.10

#### 17.3 Link Power Budget

A-29

### Little Joe Functional Specification

A link budget is useful for evaluating design decisions. The link budget predicts the operating margin, which is the amount by which the received signal level exceeds the level required to achieve a sufficiently low error rate<sup>4</sup> for a large-enough fraction of users.

The basic link equation

is:

$$P_R = P_T + G_T + G_R - L$$

where the variables, described in *Link budget terms* are in dBm or dB and  $L$  represents  $L_I$  or  $L_O$  depending on the panel location.

The operating margin  $M$  is the amount by which  $P_R$  exceeds the receiver sensitivity

$S_R$ :

$$M = P_R - S_R$$

This operating margin is calculated separately for the downlink (panel to client) and uplink (client to panel). Both margins must be positive for a client to obtain service. Since the path loss is time and location-dependent (due to fading), the fraction of users within the coverage area that have positive operating margins will vary.

The link budgets use statistical models. Their purpose is to examine the sensitivity of the system performance to design changes. It is not designed to predict performance in a specific installation. Other techniques that make use of site-specific data are used for that purpose.

term	description
$P_T$	transmitter power
$G_I$	transmitter antenna gain
$G_R$	receiver antenna gain
$L_O$	mean path loss for outdoor-indoor case
$L_I$	mean path loss for indoor-indoor case
$M$	margins for shadow and Rayleigh fading
$P_R$	received power
$S_R$	receiver sensitivity

<sup>4</sup>In our case, a frame error rate (FER) of less than  $8 \times 10^{-2}$  for 1024-byte frames

A-30

Table 6 *Link  
bud  
get  
ter  
ms*

In

#### Link Budget Parameters

we describe each of the above parameters and identify known values. In Characteristics of Conventional 802.11b Equipment

we present the radio characteristics of typical 802.11b equipment. In Path Loss Models

#### **1.11 we suggest models for the path loss and finally in**

Lin, we compute link margins and estimate the range increase of LittleJoe compared to the theoretical limits of existing APs.

#### **1.12**

#### **17.3 Link Budget Parameters**

##### **17.1 Panel Transmit Power: FCC EIRP Power Limits**

The FCC limits transmitter power for in the unlicensed 2400 to 2483.5 MHz band to 30 dBm (1 W). In addition, the EIRP for point-to-multipoint devices is limited to 36 dBm. The EIRP for point-to-point devices is not limited, but must be reduced by 1 dB for every 3 dB of antenna gain above 6 dBi.

For example, a point-to-point system using an antenna with again of 29 dBi would be restricted to an EIRP of:

$$30 \text{ dBm} + 29 \text{ dB} - (29-6)/3 \text{ dB} = 51.3 \text{ dBm}.$$

A-31

### Little Joe Functional Specification

A similar reduction in transmit power is not required for point-to-point systems in the 5725--5850 MHz band. In other words, for those client cards operating in this band (such as 802.11a clients), there is only a transmitter power limit of 30 dBm and no EIRP limit.

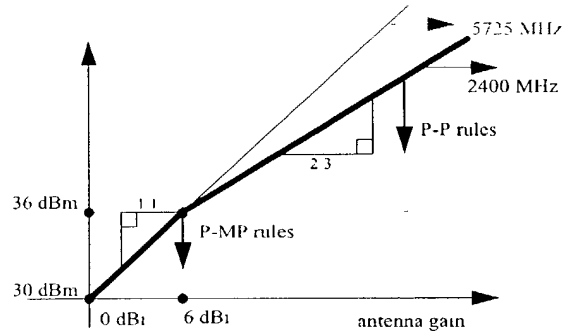


Figure 11 Diagram showing EIRP limits as per CFR 47, Part 15.247(b). Point-to-multipoint (P-MP) systems must operate below 36 dBm EIRP regardless of antenna gain and may not exceed 30 dBm transmitter power. Point-to-point (P-P) systems may increase EIRP above 36 dBm by 2 dB for each 3 dB increase in antenna gain.

### 17.3 Characteristics of Conventional 802.11b Equipment

Specifications for transmit power, receiver sensitivity and claimed indoor range were obtained from the data sheets for various manufacturer's 802.11b WLAN APs and client cards. The results are given in *Transmit power levels, receiver sensitivities and claimed indoor range*.

Sensitivities for Orinoco and Prism III measured at  $10^{-5}$  BER which is  $8 \times 10^{-2}$  FER for 1024-byte frame. Range estimates are at 11 Mb/s. Orinoco range estimate is for a *closed* environment. Apple range estimate is for *typical use*. Ericsson range quoted for an external antenna and an *office environment*.

Make/Model	transmit power (dBm)	sensitivity 1 Mb/s (dBm)	sensitivity 11 Mb/s (dBm)	indoor range (m)
Orinoco World PC card	15	-94	-82	25
Intersil Prism III	?	-91	-84	37
Apple Airport	15	?	?	45
Nokia C110/C111 client & A032 AP	15	?	-84	20-100
Ericsson PC Card PA11	20	-90	-84	75

A-32



Exhibit A-33

Intel Pro/Wireless 2011B AP	18-20	-90	-83	30
Intel Pro/Wireless 2011B client	14-18	-87	-81	30
Cisco 350	20	-94	-85	40

A-33

## Little Joe Functional Specification

Table 7 *Transmit power levels, receiver sensitivities and claimed indoor range for some conventional WLAN APs and client cards.*

### 17.1 Panel Antenna Gain

The actual panel antenna array gain is highly dependent on the implementation. However, an estimate can be obtained from its physical size and the antenna type. The antenna gain is related to its effective aperture

by:

$$G_R = \frac{4\pi A_{eff}}{\lambda^2}$$

Assuming  $A_{eff}$  is equal to the array's cross-sectional area (typical for a linear array with a

reflector):

$$G_R = \frac{4\pi wh}{\lambda^2}$$

where  $w$  is the width of the antenna,  $h$  is the height of the antenna and  $\lambda$  is the wavelength.

$$G_R = \frac{4\pi \cdot 8\lambda \cdot 4\lambda}{\lambda^2} = 128\pi = 26 \text{ dBi}$$

For the indoor unit:  $w=8\lambda$  and  $h=4\lambda$  and hence:

$$G_R = \frac{4\pi \cdot 8\lambda \cdot 8\lambda}{\lambda^2} = 256\pi = 29.1 \text{ dBi}$$

For the outdoor unit:  $w=8\lambda$  and  $h=8\lambda$  and hence:

Increasing the frequency to 5725 MHz increases the gain of a equal-sized antenna by  $(5725/2400)^2$  or about 7.5 dB (but also increases free-space loss by an equal amount).

#### 1.12.1

### 17.1 Gain Reduction due to Scattering

Computations of antenna gain and sidelobe level assume a single plane wave front arriving at the antenna. In typical WLAN installations the signal will arrive via multipath scattering and there will be many angles of arrival. This will result in a reduction in gain because signals arriving from directions other than a beam's boresight angle do not sum coherently.

The exact degree of the gain reduction depends on the antenna pattern and the angle of arrival distribution. The resulting reduction in gain can be significant. For example, in L. Greenstein and V. Ercg, "Gain Reductions Due to Scatter on Wireless Paths w/ median gain reductions of 3 to 5 dB were observed for a 37 degree half-power beamwidth antenna at 3m heights in a suburban scattering environment. Since our antenna has a narrower beamwidth and the indoor environment exhibits more severe scattering, the gain reduction may be significantly larger.

On the other hand, M. J. Gans, R. A. Valenzuela, J. H. Winters, and M.J. Carloni, "High Data Rate In reports that in most cases over half of the energy is contained in a single narrow angle of arrival. Similar results showing several widely-separated but discrete angles of arrival are visible in the data reported in G. German, Q. Spencer, L. Swindlehurst, and R. Valenzuela, "Wireless Indoor Ch and J. G. Wang, A. Mohan, and T. Aubrey, "Angles-of-Arrival of Multipath Signals in.

The "separability" of the paths in M. J. Gans, R. A. Valenzuela, J. H. Winters, and M.J. Carloni, "High Data Rate In might be accounted for by a factor of 8 difference in frequency (19GHz/2.4 GHz, which is a difference of 64 in far-field distance) and the brick interior wall construction of the building tested that increases the contribution of diffraction (single-direction) as compared to transmission and scattering (many-direction) effects.

Another effect that will reduce the gain of the antenna when there are nearby scatterers is that the wave fronts are not well approximated by plane waves and this results in an additional loss of gain and increase in sidelobe ratio.

In the link budgets we have entered some arbitrary values for the gain reduction. These values are to be refined through propagation measurements.

#### 1.12.2

##### 17.1 Client Antenna Gain

Amongst those vendors considered, only Ericsson provides a gain specification for their PC client card antenna (0 dBi). This number agrees with measurements reported by others for other client cards. Omnidirectional antennas used by enterprise APs have higher gains. For example, the Nokia C950 has a gain of 2.5 dBi and the Cisco AIR-ANT3213 has a gain of 5.2 dBi.

##### 17.1 Client and Panel Noise Figures

The Intersil Prism II receiver IC specifies a noise figure of about 2 dB. Losses in the antenna switch and bandpass filter will increase this number, perhaps to 4 dB. The Orinoco "Ruby" receiver IC has a noise figure of about 5 dB.

A-35

## Little Joe Functional Specification

The LittleJoe panel uses a 1 dB NF LNA and a circulator with 1 dB loss to achieve a NF of about 2 dB.

A client card chipset with an external LNA is used in LittleJoe. The difference between typical client and predicted LittleJoe panel noise figures must be included in the link budget. This difference is also used when comparing the performance of the LittleJoe panel with conventional APs.

### 17.1 Effect of Shadow Fading

#### 17.3 The path loss models described Path Loss Models

estimate the median path loss for a given distance. However, different locations with the same path distance will have different path losses. These variations have been found to be normally distributed when the path loss is expressed in dB. The standard deviation of the path loss depends on the scattering environment but typical values for office environments are 3 to 6 dB.

We cannot increase the transmitter power to compensate for shadow fading since the system is already operating at maximum transmit power levels. Instead, the shadow fading reduces the fraction of the coverage area that can be serviced.

We will assume the coverage area is a circle or a “wedge” of a circle so that we can use the equations derived for circular cells W.C. jakes, ed.,

$$a = \frac{-\gamma}{\sigma\sqrt{2}}$$

$$b = \frac{10n\log e}{\sigma\sqrt{2}}$$

$$F(\gamma) = \frac{1}{2} \left( 1 - \operatorname{erf}(a) + e^{\left(\frac{1-2ab}{b^2}\right)} \left( 1 - \operatorname{erf}\left(\frac{1-ab}{b}\right) \right) \right)$$

:

where  $\gamma = M$  is the link budget margin at the coverage boundary (dB),  $\sigma$  is the standard deviation of fading (dB), and  $n$  is the path loss exponent. For other coverage region shapes, a different expression must be derived or the value computed through numerical integration.

The percentage coverage computed above is an average over the whole coverage area and it may include large coverage “holes.”

A-36

Since shadow fading is caused by objects such as walls, bookcases, doors, etc. we should expect the dimensions of the “holes” to be approximately the same as the dimensions of the shadowing objects. This is unlike the Rayleigh fading where the fades have dimensions on the order of the wavelength.

### 1.12.3

#### 17.1 Effect of Rayleigh Fading

A margin is usually included in a link budget to counter the effect of multipath fading. For NLOS propagation this fading is Rayleigh-distributed. The probability that a Rayleigh distributed random variable  $r$  will be  $R$  dB below the mean can be approximated

by:

$$P(r < R) = 10^{-R/10}$$

for  $R < 01$ . For example, the signal will be 10 dB below the mean about 10% of the time and 20 dB below the mean about 1% of the time.

For typical indoor scatterers, the duration of the Rayleigh fades (tens or hundreds of milliseconds) is slow relative to the frame duration (less than about 10 milliseconds).

The fading on antennas separated by a significant fraction of one wavelength is weakly correlated. Many clients use switching diversity to combat Rayleigh fading. The client's receiver switches between two antennas until it finds a signal that is sufficiently strong. This squares the probability of fading to  $10^{-2R/10}$ .

Rayleigh fading affects data throughput rather than coverage. Not including any fade margin would result in the channel being unavailable about 40% of the time without diversity and about 16% of the time with two-antenna diversity. Allowing a 10 dB Rayleigh fading link margin would mean the signal was faded about 10% of the time without diversity and about 1% of the time with two-antenna switched diversity. Allowing a 5 dB fade margin, would mean 10% probability of fading with two-antenna switched diversity.

The effect of Rayleigh fading on data throughput is difficult to compute because of complex interactions between frame loss and contention-control mechanisms in the 802.11 MAC and congestion-control mechanisms in TCP/IP.

As far as the link budget is concerned, we assume a 5 dB fade margin for downlink transmissions. This assumes that the clients have switched diversity. We also assume a 5 dB fade margin for the uplink since the multiple antenna elements will provide some diversity gain against fading.

A-37

Little Joe Functional Specification

### 17.3 Path Loss Models

This section describes the models used to predict the outdoor-indoor and indoor-indoor path loss. Descriptions and experimental validation of these path loss models can be found in J. Kivinen, X. Zhao, and P. Vainikainen, "Empirical Characterization of W1

#### 17.1 Propagation in Free Space (LOS)

Loss in free space is given

$$L_{FS}(d) = 20 \log \left( \frac{4\pi d}{\lambda} \right)$$

by:

in the far field ( $d > 2D^2/\lambda$ ,  $d \gg D$  and  $d \gg \lambda$  where  $D$  is the largest dimension of the antenna).

#### 1.12.4

#### 17.1 Propagation by Diffraction (NLOS)

For NLOS (non line-of-sight) paths, propagation is mainly by diffraction and a power-law path loss formula is a good model:

$$L(d) = L_{FS}(d_0) + 10n \log \left( \frac{d}{d_0} \right)$$

where  $d_0$  is a reference LOS distance (free space break point) and  $n$  is the path loss exponent a value that depends on the geometry of the paths. For indoor NLOS paths  $n$  is typically between 3 and 4.

#### 1.12.5

#### 17.1 Propagation by Transmission (OBS)

When propagation is mainly by transmission through walls for floors, a simple model is to modify the free-space path loss with a wall or floor attenuation factor for each penetrated wall and/or

floor:

$$L_{OBS}(d) = L_{FS}(d) + p_{wall} W_{wall} + p_{floor} W_{floor}$$

where  $p_{wall}$  and  $p_{floor}$  are the number of penetrated walls and floors respectively and  $W_{wall}$  and  $W_{floor}$  are attenuation constants that depend on the wall and floor construction materials.

A-38

For the same-floor propagation through several walls, this model can be simplified by substituting a constant attenuation per unit distance:

$$L_{OBS}(d) = L_{FS}(d) + \alpha d$$

where  $\alpha$  is the attenuation per meter. A typical value for  $\alpha$  in buildings with hard walls is 0.6 dB/m (resulting, for example, from  $W_{wall}$  of 4 dB per wall and a wall every 6 or 7 meters).

#### 1.12.6

### 17.1 Outdoor-Indoor Path Loss

A model for outdoor-to-indoor propagation J.E. Berg, "Building Penetration Loss along Urban Street Microcells," in and E. Damosso and L. Correia, eds., combines the OBS and LOS models and a correction for the angle of incidence:

$$L_O(d) = L_{FS}(S + d) + WG_e \left(1 - \frac{D}{S}\right)^2 + \max(\Gamma_1 + \Gamma_2)$$

where  $S$ ,  $d$ , and  $D$  are in meters (see *Model for COST 231 building penetration loss model*. From E. Damosso and L. Correia, eds , ),  $W_e$  is a constant related to the external wall construction (about 7dB for concrete walls with unshielded windows) and  $WG_e$  is a similar constant for shallow angles of incidence (20 dB)

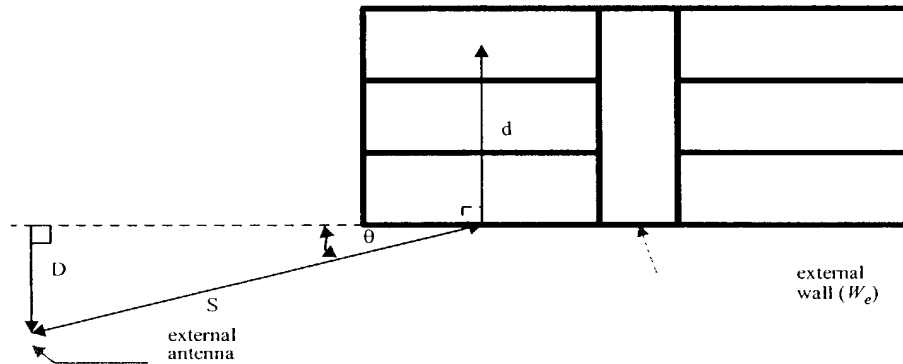


Figure 12 *Model for COST 231 building penetration loss model*. From E. Damosso and L. Correia, eds ,

A-39

## Little Joe Functional Specification

The loss inside the building is modeled

using:

$$\Gamma_1 = W_i P$$

or

$$\Gamma_2 = \alpha(d-2)\left(1 - \frac{D}{S}\right)^2$$

where  $W_i$  is the additional loss per interior wall (4 dB per wall),  $p$  is the number of walls passed, and  $\alpha$  is the attenuation constant, about 0.6 dB/m.

If there is significant building penetration from several directions, the signal levels from each direction should be computed separately and summed. Note that the building penetration loss  $W_e$  may be 10 to 20 dB higher for buildings with low-emissivity (energy-efficient, “low-E”) windows that have metallic coatings.

## 1.12.7

### 17.1 Indoor-Indoor Path Loss

The power-law NLOS model described in Section 4.4.2 is usually used to model indoor-to-indoor path loss.

## 1.13

### 17.3 Link Budget Results

[5] A series of Link Budget spread sheets have been prepared and are available. It is important to note that depending on assumptions about the deployment scenario, the range estimates may vary widely. Hence, in this section, we calculate the link margin gain of the LittleJoe panel compared to conventional APs (transmitting at 30 dBm) and report the resulting increase in range in *Estimated range increase for indoor and outdoor deployments co.* For results based on different deployment environments, see Siavash Alamouti, LittleJoe Link Budget Spread Sheets, May 10, 2002

, Ed Casas, LittleJoe Beamforming, .

	indoor unit		outdoor	
	downlink	uplink	downlink	uplink
Theoretical transmit antenna gain	26.0 dB <sub>i</sub>	0 dB <sub>i</sub>	29.1 dB <sub>i</sub>	0 dB <sub>i</sub>
Required backoff	6.7 dB	0 dB	7.7 dB <sub>i</sub>	0 dB

A-40



# WAI

## Little Joe Functional Specification

Table 8 *Estimated range increase for indoor and outdoor deployments compared to omnidirectional APS.*

The FCC limits transmitter power in the unlicensed 2400 to 2483.5 band to 1 Watt (or 30 dBm). However, the transmit power has to be backed off by 1 dB for every 3 dBi of antenna gain over 6 dBi. The approximate<sup>5</sup> effective transmit gain for indoor and outdoor deployments are calculated in *Estimated range increase for indoor and outdoor deployments co.*

The added coverage compared to regular APs is a function of the link margin above regular APs shown in *Estimated range increase for indoor and outdoor deployments co.* The best-case range increase is based on a path-loss exponent of 2, the worst-cases is based on path loss exponent of 4 and the nominal path loss exponent was set to 3.5 indoors and 3 outdoors. Note that the results in *Estimated range increase for indoor and outdoor deployments co* indicate that the range increase is more significant on the uplink. This is due to the full antenna gain on the uplink (no transmit back-off as in downlink) and the NF improvement in the LittleJoe receiver. Some of the gain is offset by the assumption that a 4 dB windowing and pointing loss is incurred on the uplink. This is calculated based on windowing SNR loss of 1.5 dB SNR on the uplink and a pointing loss due to lack of fine steering on the downlink and uplink.

In *Estimated range increase for indoor and outdoor deployments co*, we have used the actual transmit and receive antenna gains based on measured gains in our laboratory. The actual gain for the half height antenna was measured to be around 24 dBi (compared to theoretical value of 26 dBi), and for the full height antenna at around 27 dBi (compared to 29.1 dBi).

[5] In practice, the 802.11 clients transmit significantly lower power than the FCC limits (15dBm to 17 dBm). Therefore, there would be significant disadvantage in the uplink. The comparison in *Estimated range increase for indoor and outdoor deployments co* is based on the limits of our technology compared to omni-directional equipment. For actual range estimates with existing clients please refer to Siavash Alamouti, LittleJoe Link Budget Spread Sheets, May 10, 2002

## 5. Transmit Power Control

LittleJoe does not perform transmit power control on a per-client basis. This is to be compliant with the MAC channel sense mechanism. Per-client power control may hide the transmission to one client from the other clients in the network. This would create a hidden beam problem. Therefore, the power control scheme for LittleJoe is applied to all the radios and is independent of the beam, channel, destination (client) or frame type. The power is reduced if and only if all

<sup>5</sup>The effective antenna gain and hence the resulting back-off depend on the implementation and may be slightly different

A-42

the associated clients have an SNR of more than 30 dB. Otherwise, the maximum power is applied.

#### 1.14

### 17.3 Power Control Requirements

The term “maximum level” used in the following refers to the maximum transmit power per antenna that is allowed by FCC.

The transmit power level.

- shall never exceed the maximum level
- shall be accurate to within 1.0 dB with probability 99.999%
- shall be set with a resolution of 0.5 dB or better<sup>6</sup>
- shall be set to the maximum level if the SNR of any associated client is less than 30dB
- shall be reduced by 1 dB for every 1 dB that the minimum SNR of any associated client<sup>7</sup> exceeds 36 dB
- shall be able to be reduced by at least 24 dB from the maximum level at 6 dB steps (to within 2.0 dB accuracy)
- is raised to the maximum level for a duration of 10 seconds immediately<sup>8</sup> following reception of a probe request frame

The relationship between (downlink) transmit power level and minimum (uplink) SNR is shown in *Transmit power as a function of me*. The figure shows that the transmit power is set according to the power requirements of the client with minimum SNR.

---

<sup>6</sup>The accuracy and resolution of power control should be included in the link budget as a 1.5 dB loss for downlink transmissions.

<sup>7</sup>Stations from which no frames have been received in the last 24 hours are expected to be disassociated

<sup>8</sup>Since the transmit power is controlled by the host, there may be significant delays in changing the power level. One or more of the initial probe responses may be transmitted at the initial (lower) power levels. If the client is unable to receive any of the probe responses (e.g. because they were transmitted at the initial power level), it will scan again and will then receive the probe responses transmitted at the higher power level. It is possible that in some cases the client will receive some responses at the initial level and some at the higher level and choose the wrong beam as a result. This situation will be handled by the roaming algorithms

A-43

## Little Joe Functional Specification

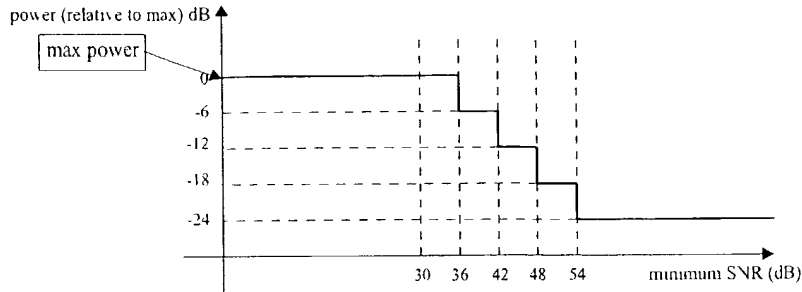


Figure 13 Transmit power as a function of measured uplink SNR

### 17.3 Power Control State Transition Diagram

Figure 14 The power control state machine for each panel is shown in *The power control state machine*

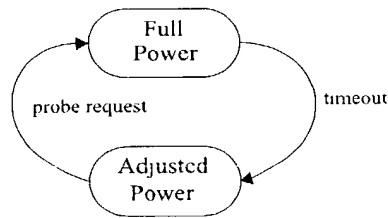


Figure 14 *The power control state machine*

The state machine can be in one of two states:

- **Full Power:** the transmit power is set to the maximum level. This state is required to allow all clients to receive probe responses and beacons transmitted from the panel.
- **Adjusted Power:** the transmit power is adjusted using the SNR of the client with the lowest SNR (see the rules above). This is the normal operating mode and provides some power control with enough margin so as not to impact performance.

The rules governing the state transmissions are:

A-44

- *timeout*: the time allowed for associations to complete (e.g. 10 seconds) has passed since the most recent probe request
- *probe request*: a probe request frame is received on any beam

**Important Note:** *this design requires APs to indicate received probe request frames to the host (even during normal operation) If this is not possible, full transmit power should be applied at all times.*

## 1.15

### 17.3 Power Control SNR Measurement

The client SNR estimate is a time-averaged value. Let the SNR measurement for the last frame be denoted by  $SNR_{new}$ . The average SNR is calculated

$$\overline{SNR} = 0.9\overline{SNR} + 0.1SNR_{new}$$

as:

## 6. Media Access Control (MAC)

The LittleJoe MAC subsystem is composed of a host interface, thirteen MAC controller chips, a Multi-MAC controller (MMC) and thirteen baseband (modems) attached to thirteen radios. Each of the MAC controller chips are IEEE 802.11b based using standard 802.11 DCF (CSMA/CA) as the mechanism for scheduling transmissions. Each MAC controller chip independently monitors a different beam to determine the next opportunity for transmission. Once one of the MAC controller chips transmit, the other MAC controller chips on the same channel sense the transmission and wait their turn to transmit. This approach guarantees that each MAC controller/radio has an equal chance to transmit. Additionally, the MMC helps avoid collisions of downlink transmissions with ongoing uplink receptions.

## 1.16

### 17.3 ViVATO Panel MAC Operation

## 1.17

As described in , the LittleJoe architecture has one WLAN radio on each Butler Matrix port. Each radio operates with its own independent MAC controller.

A-45

Little Joe Functional Specification

All of the radios listen simultaneously but only one of them can transmit at a time. Each radios's MAC protocol enforces the "one transmitter at a time" rule through the use of it's CCA circuitry. Any transmission from any radio on any beam will interfere with and cause the loss of any frames currently being received on other beams on the same channel. However, since the radios on different beams can hear each other's transmissions, there should be no collisions after the initial downlink DATA or CTS frame (as a result of MACS on other beams setting their NAV timers). However, an MMC is added to avoid collision of downlink and ongoing uplink packets. The details of the MMC function is described in

Mul.

The LittleJoe panel supports unmodified 802.11b clients. No special MAC software is required at the client. Each beam appears as a different AP and a client associates with whichever beam it thinks is providing the best coverage. However, since the clients are portable and the wireless environment may change, the initially selected radio may not indefinitely be the best radio and hence the client may have to roam to another radio. To support a client transition from one BSS (radio) to another BSS (radio) within the same ESS, there needs to be a mechanism to support 802.11 MAC layer mobility. Inter-access point communication (IAPP) specified in the IEEE 802.11f draft supports that mechanism. The details of the roaming function is described in

Intr.

Since different APs in the panel may be assigned to different channels, a channel assignment algorithm is used as described in

A-46

Cha.

A traffic-shaping functionality described in Section 10 is implemented to limit the downlink load to ensure system stability and fairness between uplink and downlink traffic loads.

#### 1.17.1

##### 17.1 MAC Modes of Operation

Currently the IEEE 802.11 standard provides two modes of operations which are all part of the basic service set (BSS), which is defined as a group of clients that communicate with each other. The two modes are Independent basic service set (IBSS), and infrastructure basic service set (BSS).

The Little Joe panel will only support the infrastructure BSS mode.

Infrastructure BSS mode of operation is distinguished by the use of an access point (AP). The AP is used for all communications including communication between clients in the same service set.

If one client transfers a frame to the second client, it must take two hops. First, the originating client transfers a frame to the AP. Second, the AP transfers the frame to the client. With all communications relayed through the access point, the service area is defined by the coverage area of the AP itself.

In infrastructure BSS mode, clients must associate with an AP before communication can begin. The client always initiates the association process and the AP may choose to grant or deny service. Associations are exclusive in that a client can only be associated to one AP at a time.

LittleJoe has 13 independent 802.11 APs within one panel. We hereby refer to the APs within the panel as radios. The APs need to be coordinated to form a single entity. This is supported by the Extended Service Set (ESS) in 802.11. For a LittleJoe panel to properly cover a 100 degree area with portability support, there is a requirement for the union of multiple BSSs to be configured to be part of the same ESS. This requires that the 13 radios operate in concert to allow the outside world to use a single MAC address to talk to a client somewhere within the ESS. To support this functionality, the LittleJoe backbone must act as a single link-layer domain.

Clients within the same ESS may communicate with each other, even though these clients may be located on separate beams. For clients in the ESS to communicate with each other, the wireless medium must act as a single layer 2 connection. Radios act as bridges, so direct

A-47

## Little Joe Functional Specification

communication between clients in an ESS requires that the backbone network also be a layer 2 connection.

The ESS may also function between multiple LittleJoe panels. This requires that, in addition to the radios inside the panel working as a single ESS, the Ethernet backbone may also be used as the backbone for the ESS.

## 7. Multi-Radio Transmit Control

The panel has 13 independent radios that are coordinated through the CSMA protocol. The protocol tries to ensure that only one transmitter per channel operates at a given time. However, there still remains a small chance of a transmit collision. Therefore, to avoid exceeding FCC transmit power limits and slight degradation in performance, a transmit control function is provided. The control function ensures that:

- only one transmitter transmits at a given time on any given channel (first-come, first-serve)
- only complete frames are transmitted (i.e.; frames that begin while another transmission is in progress are discarded)
- if not possible to determine which of several transmissions occurred first, then select any of the them arbitrarily.

### 17.3 Multi-MAC Control (MMC)

LittleJoe uses 13 radios. Without any coordination between these radios, transmitting a downlink packet on one radio would destroy any uplink packets being received simultaneously on another radio. This effect is referred to as “data suicide”. Allowing each radio to operate independently causes serious performance issues LJ MAC problems: Using multiple co-located APs with smart antennas, A Tu. MMC controls the transmissions of each radio to prevent data suicide. More details on MMC are discussed in Little Joe Multi-MAC Controller, (aka CCA Glue Logic), Presentation/Design re.

1.18

### 17.3 MMC Overview

Figure 15 The MMC operation is shown in *LittleJoe Multi-MAC Operation*

48

A-48



There are 13 radios with independent CCA input and outputs. For clarity, we call these *busy\_out* and *busy\_in*. The *busy\_out* indicates the state of the CCA output as detected by the baseband processor and *busy\_in* indicates the state of the CCA input to the MAC processor. The *busy\_out* signals are fed into the MMC. Based on the values for *busy\_out* signals, the channel assignment vector, and the *busy\_out* enable vector, the MMC sets the *busy\_in* signals for every radio.

The MMC function is as follows:

Define a signal, *global\_busy*, for each channel. *global\_busy* is active if:

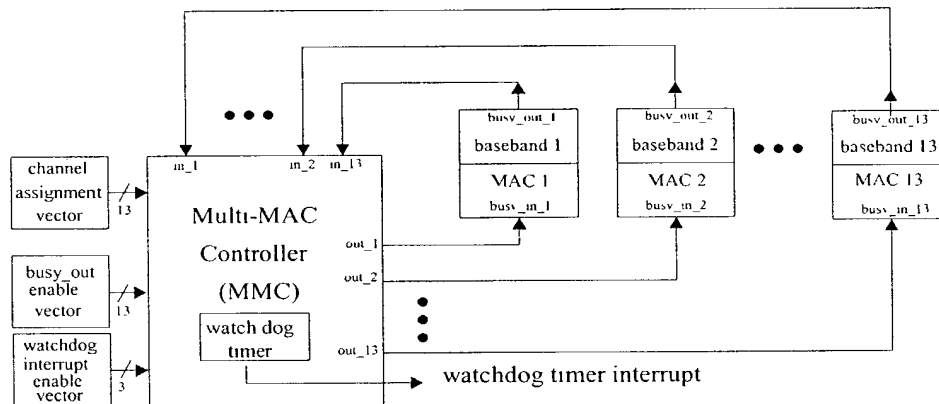
- *busy\_out* for any radios operating on that channel indicates busy excluding those radios whose *busy\_out* enable signal is not set.

Set the MMC output for a given radio to busy if:

- *busy\_out* for that radio indicates busy, or
- *global\_busy* for this radio's channel is active.

Each of three watchdog timers monitors the period that *global\_busy* has been active for that channel. If *global\_busy* has been active for more than *aBusyActiveThreshold*, and that interrupt is not disabled (indicated by the watchdog interrupt enable vector), an interrupt is generated to the host.

The *busy-out* enable vector is set taking into consideration the effect of interference and overlapping BSS. It ensures that external channel activity does not quiet all the radios in the panel.



A-49

Little Joe Functional Specification

Figure 15 *LittleJoe Multi-MAC Operation*.

### 17.3 MMC Provisioned Parameter

The only provisioned parameter for the MMC is:

- `aBusyActiveThreshold`: the threshold of *global-busy* active status beyond which an interrupt is generated to the host.

## 8. Intra-Panel Roaming

The LittleJoe switch contains 13 radios operating in the AP mode. A client initially associates to one radio. It selects the radio with the best signal at the association time. However, since the clients are portable and the wireless environment may change, the initially selected radio may not indefinitely be the best radio and hence the client may have to roam to another radio.

However, roaming is initiated by clients and cannot directly be controlled by the radios in the ViVATO switch. The roaming behavior is dependent on the client implementation. In most commercially available clients, roaming is triggered when the channel quality (SNR) falls below a threshold. The channel quality assessment (SNR measurement) is based on received beacon strength. To ensure that a client is associated with the best radio, ViVATO's switch forces the client to roam to the radio with the best signal quality. This is done using the beam-switching algorithm.

Also, to ensure seamless roaming between beams as directed by the client, we support the Inter-Access Point Protocol (IAPP) which is an extension to 802.11 to support interoperability, mobility, handover, and coordination between APs (or radios) in a wireless LAN.

For LittleJoe to support roaming or load balancing for a client, it is necessary to support IAPP where reassociation occurs in the MAC layer and is transparent to the upper layers.

The specification of IAPP is defined by IEEE 802.11f and this specification is used as a baseline for the implementation of handover messaging between beams. The messaging is implemented within the panel's host controller. However, at a later time, it may be viable to expose the IAPP messaging through the backhaul to other LittleJoe panels as well as other 802.11f compliant third party APs.

### 17.3 Roaming Requirements

A-50

- Beamswitching: To ensure that the clients are associated with the radio (beam) with the best signal level
- IAPP: to ensure client-initiated seamless roaming between the radios in the panel.

## 1.19

### 17.3 Beam-Switching Algorithm

The roaming algorithm disassociates the client once it moves out of the associated main beam. However, such movement is difficult to detect in the wireless environment and disassociation may result in packet loss and long association procedure. The effect is particularly significant for clients located between two neighboring beams. So, the roaming algorithm will disassociate the client when there is significant difference between signal qualities on different beams.

#### 1.19.1

### 17.1 Beam-Switching State Transition Diagram

Figure 16 As shown in *The roaming state machine*.

A-51

# Little Joe Functional Specification

, the roaming state machine for each client served by the panel has three states:

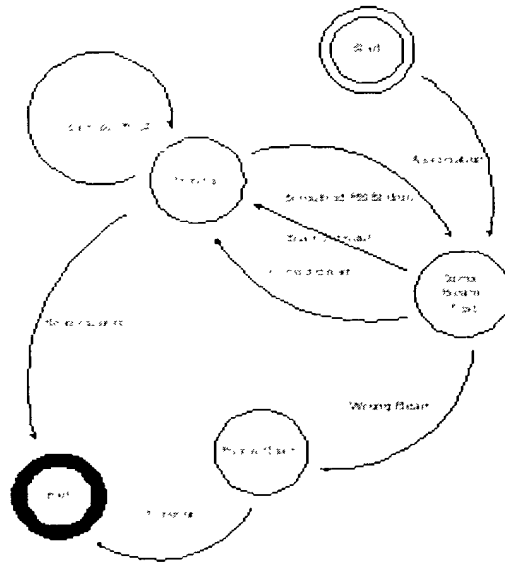


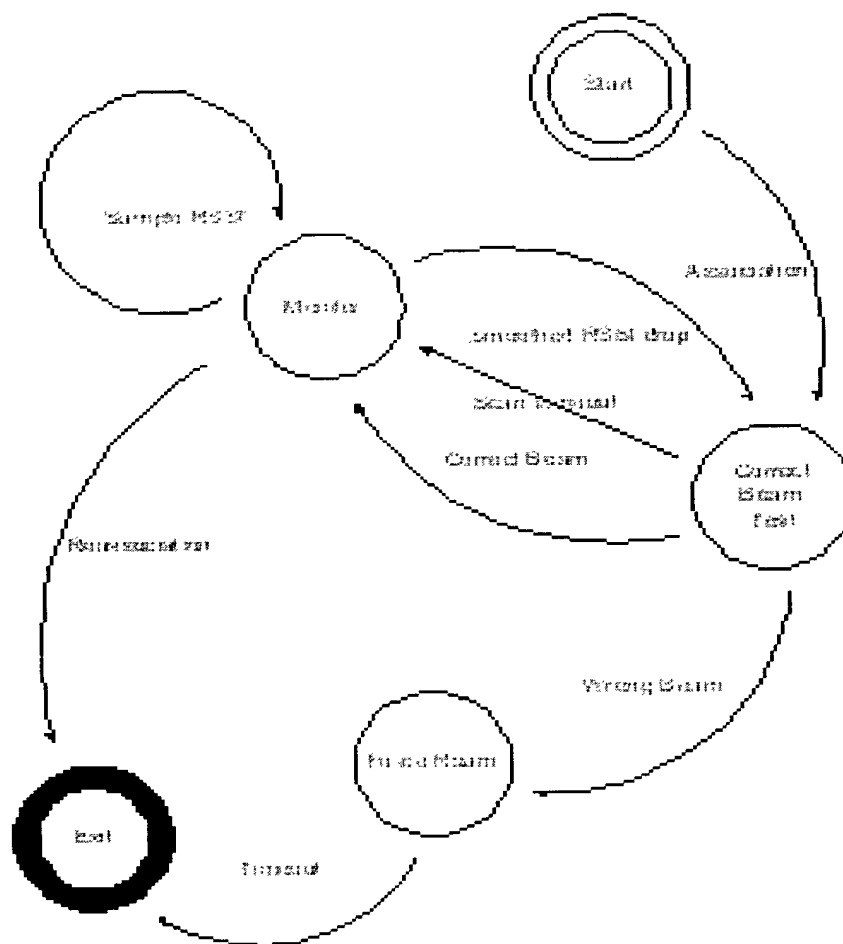
Figure 16 The roaming state machine

**Monitor:** once a client is associated to a beam, the radio continues to collect RSSI values for each packet received from that client. It recalculates a new measure called the *aSmoothedRSSIValue*<sup>9</sup> over a window size of *aRSSIWindowSize* and compare it to a threshold called the *aRSSILowerControlLimit*.

**Correct Beam Test:** the scanning radio is used to measure the RSSIs and calculate *aSmoothedRSSIValue* for the client on each of the adjacent ports. *aRSSIWindowSize* samples for the two adjacent ports are averaged and compared to the same parameter for the current beam to determine the best beam.

<sup>9</sup>The roaming algorithm has no provisioned parameters. All the parameters are internal to the algorithm and may not be altered by Network Management. In this document all internal parameters use the oblique style of the AvantGarde font (*aInternalParameter*) and provisioned parameters use the Arial font (*aProvisionedParameter*).

A-52i



# A-5Zii

**Force Roam:** the client is placed temporarily onto a black list so that it cannot associate to the current beam. Then the dissociation procedure will be called.

The rules governing state transitions and the subsequent actions are:

- *Association:* this transition occurs automatically when the client associates
- *Sample RSSI:* recalculate  $aSmoothedRSSIValue$  and  $aRSSILowerControlLimit$ .
- *Smoothed RSSI drop:*  $aSmoothedRSSIValue$  drops to  $aRSSILowerControlLimit$
- *Correct Beam:* scan indicates current beam is the best.
  - Action: resample with new RSSI values and recalculate a new  $aLowerControlLimit$
- *Roaming Scan Timeout:* the scanning radio has been monitoring the neighbouring beams for more than  $aRoamingScanTimeout$  without any decision about the correct beam.
- *Wrong Beam:* scan indicates a better beam whose RSSI exceeds the RSSI of the current beam by  $aSignalDropThreshold$  dB
  - Action: black-list the client and force a disassociation.
- *Roaming Timeout:* Timeout after  $aRoamingTimeOut$ .
  - Action: remove the client from the black list and remove any state information about the client.
- *Reassociation:* client initiates a reassociation to a new beam.
  - Action: allow the reassociation sequence to take place and remove state information about the client.

#### 8...1. Calculation of $aLowerControlLimit$

The parameter  $aLowerControlLimit$  is calculated using both the mean and the standard deviation of RSSI. The  $aLowerControlLimit$  is calculated as follows:

$$aLowerControlLimit = \overline{RSSI} - 2\sigma$$

$$\overline{RSSI} = \frac{1}{N} \sum_{i=0}^{N-1} RSSI_i, \quad N = aRSSIWindowSize \text{ in frames}$$

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (RSSI_i - \overline{RSSI})^2}$$

where  $RSSI_i$  is the RSSI value reported for frame  $i$ . The  $N-1$ th frame is the most recent frame.

#### 8...2. Smoothed RSSI Calculation

A-53

#### Little Joe Functional Specification

To detect client movement, the recommendation is to sample RSSI values continuously as a packet arrives and to calculate *smoothedRSSIValue* (S). This can be calculated with the following formula.

$$S_j = 0.1RSSI_j + 0.9S_{j-1}$$

This value is then compared to the *LowerControlLimit* and if it is larger than the limit, the client enters the Correct BeamTest state.

#### 1.20

### 17.3 IAPP (Seamless Roaming)

The objective of IAPP is to enable seamless client-initiated roaming between beams within the panel, between panels, and between a panel and third party APs.

Seamless roaming enables clients that are associated with the following features:

- Clients radios are able to transition across beams within a time resolution of about 100 ms without user intervention.
- No intervention required by the operating system. Therefore independent of operating system used by the client.
- Roam across multiple panels
- Roam from the ViVATO panel to 3<sup>rd</sup> party APs that conform to IEEE 802.11f.

For details of IAPP, please refer to Jim, Brennan, Seamless Roaming for LittleJoe, Internal ViVATO Publication, Se.

## 9. Channel Assignment

The DP2330 channel assignment includes two parts: measurement of metrics and channel assignment.

The metric measurement function resides in the host. It measures the channel activity and provides information for channel assignment and other purposes. Channel assignment function is provided in the management software package. It provides the best channel assignment based on the given measurement information. The management software functions related to channel assignment is decided by software development. This section only focuses on the algorithm.

A-54

## 1.21

### 17.3 Channel Assignment Provisioned Parameters

- *aChannelAssignmentCycle*: the time duration between changes in the channel assignment (default value 24 hours).
- *aHeavyInterference*: the interference activity threshold. If interference activity is above this value, the channel is considered as **Bad Channel** (value TBD).
- *aBadChannelThreshold*: the number of measurement periods (*aMeasurementDuration*) that a channel has interference activity above *aHeavyInterference* threshold (default value 4).
- *aJamInterference*: the interference activity threshold. Interference activity is above this value in the last measurement, **Emergency Exit** is triggered (value TBD, *aJamInterference* > *aHeavyInterference*).

## 1.22

### 17.3 Channel Assignment Internal Parameters

- *aMeasurementCycle*: the time duration in which a complete measurement is done (default value 24 hours).
- *aMeasurementDuration*: the time duration between two measurement points (default value 30 minutes).
- *aPeakLoadLimit*: the maximum load allowed on one channel (value TBD)
- *aChannelSixBiasFactor*: the bias factor to penalize transmission on channel 6 to reduce the intermodulation problem.

## 1.23

### 17.3 Channel Assignment Metrics

The scanning radio and traffic radios have to measure some metrics of channel activity. The measurement shall repeat in a cycle of *aMeasurementCycle*. During *aMeasurementCycle*, the metrics are measured every *aMeasurementDuration*. The desired metrics include: number of associated clients, throughput and packet error rate (PER) of each traffic radio; interference and channel utilization of each beam/(frequency) channel. The metrics are defined below.

- $N_i(t)$ : Number of associated clients of the  $i$ th traffic radio. It is collected from the MIB, and is averaged over *aMeasurementDuration* period.
- $S_i(t)$ : Throughput of the  $i$ th traffic radio. It is in packets/second or bytes/second, whichever is available. It is also available from the MIB, and is averaged over *aMeasurementDuration* period.

A-55



### Little Joe Functional Specification

- $P_i(t)$ : PER of the  $i$ th traffic radio. It is collected from the MIB, and is averaged over  $aMeasurementDuration$  period.
- $D_i(t)$ : delay of the  $i$ th traffic radio. It is collected from the MIB, and is averaged over  $aMeasurementDuration$ .
- $p_{ij}(t)$ : channel utilization of the  $i$ th beam on the  $j$ th channel. It is the portion of time CCA is set for a given channel. It is measured by both traffic and scanning radios and is averaged over  $aMeasurementDuration$ . We refer to this as Channel Utilization Factor (CUF)
- $N_{sj}(t)$  number of downlink packets transmitted on the  $j$ th channel. It is averaged over  $aMeasurementDuration$  period.
- $Nr_{ij}(t)$  number of correctly received uplink packets transmitted by the clients associated with the  $i$ th beam on the  $j$ th channel. It is available in the MIB and is averaged over  $aMeasurementDuration$  period.
- $Nn_{ij}(t)$ . number of uplink packets transmitted by the clients associated with other beams, which are correctly received by the  $i$ th beam on the  $j$ th channel. It is measured by the scanning radio and is averaged over  $aMeasurementDuration$  period. We call this the Self Interference Metric (SIM).
- $No_{ij}(t)$  number of uplink packets transmitted by the clients from overlapping subnets, which are correctly received by the  $i$ th beam on the  $j$ th channel. It is measured by the scanning radio and is averaged over  $aMeasurementDuration$  period. We call this the Overlapping Subnet Interference (OSI).
- $Ne_{ij}(t)$ . number of uplink packets with PLCP or data CRC errors in the  $i$ th beam on the  $j$ th channel. It is measured by the scanning radio and is averaged over  $aMeasurementDuration$  period. We call this the Unidentified Interference Metric (UIM).
- $I_{ij}(t)$ : interference of the  $i$ th beam on the  $j$ th channel. It is the portion of time CCA is set due to interference. It is measured by the scanning radio.

All the metrics are maintained in a table within  $aMeasurementCycle$ . When the cycle restarts, the table can either be cleared or updated with some aging factor. The table update mechanism is TBD.

It is difficult to measure  $I_{ij}(t)$  when there are traffic radios on the same channel. In such cases,  $I_{ij}(t)$  can be derived from other measurements. To estimate  $I_{ij}(t)$  we first estimate the total number of packets from the overlapping subnets. Assuming that all downlink packet transmissions from the panel lead to CCA high, and assuming that all uplink packets have the same error probability. Then the total number of packets (with and without CRC errors) from overlapping subnets may be estimated

by:

$$NI_{ij}(t) = No_{ij}(t) + Ne_{ij}(t) \times \frac{No_{ij}(t)}{Nr_{ij}(t) + Nn_{ij}(t) + No_{ij}(t)}$$

A-56

And  $I_{ij}(t)$  may be estimated

by: 
$$I_{ij}(t) = \frac{NI_{ij}(t)}{Ns_i(t) + NI_{ij}(t) + Nn_{ij}(t) + No_{ij}(t) + Ne_{ij}(t)} \rho_{ij}$$

The metrics  $N_i(t)$ ,  $S_i(t)$  and  $I_{ij}(t)$  are necessary for the channel assignment algorithm. The remaining metrics are used to estimate  $I_{ij}(t)$  and would therefore not be needed if there was a direct way to measure interference.

## 1.24

### 17.3 Channel Assignment Algorithm

#### 17.1 Channel Assignment Preprocessing

##### 9...1. Eliminate Bad Channels

A channel that has interference activity more than `aHeavyInterference` for `aBadChannelThreshold` is not used. The interference activity is averaged over intervals of `aMeasurementDuration`. There are typically 48 measurement intervals in one `aMeasurementCycle`. If the number of periods where interference activity is more than `aHeavyInterference` exceeds `aBadChannelThreshold`, then that channel is eliminated.

##### 9...2. Estimate Total Users in the Beam

The total number of active users in the beam may be estimated by dividing the number of associated users in that beam by the percentage of time available to those users. The total number of users on beam  $i$  and channel  $j$  may therefore be described

by: 
$$N_{ij}(t) = \frac{N_i(t)}{1 - \tilde{I}_{ij}(t)}$$

where:

$$\tilde{I}_{ij}(t) = \min \{ I_{ij}(t), \text{aHeavyInterference} \}$$

which is the interference activity limited to the maximum allowable interference on a given beam. This ensures that the estimate does not provide large peaks due to unusual period of high interference.

#### 1.24.1

K-57

## Little Joe Functional Specification

**17.1 Block-based Channel Assignment Algorithm**

The block-based channel assignment algorithm assigns neighbouring beams to the same frequency channel (see Figure 1). Such assignment can help minimize the hidden beam problem.

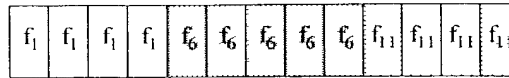


Figure 17 Figure 1 Block-based channel assignment

The algorithm breaks the 13 beams into a maximum of 3 blocks, with each block assigned to one frequency channel (channels 1, 6, or 11), so that the peak load on each channel is minimized. In order to find the optimal solution, we have to determine:

- the boundaries between the assignment blocks (i.e. the number of beams in each block),
- the frequency channel of each block.

There are a total of possible 66 combinations that divide 13 beams into 3 blocks. For each of these possible combinations, the three blocks have to be assigned to 3 different channels. The number of channel permutations is 6. So, we have to find out the best channel-beam combination from  $66 \times 6 = 396$  possible total combinations. Denote  $L_j(t)$  as the total load on the  $j$ th frequency channel at time  $t$ . Let

$$L_j^* = \max\{L_j(t)\} \quad t \in [0, T]$$

be the peak load on the  $j$ th channel in the last measurement period, where  $T$  is the measurement cycle (*aMeasurementCycle*). Our objective is to perform an exhaustive search to find the combination that minimizes the peak load on all channels. Mathematically, this may be described as:

$$\min\{\max\{L_j^*\}\} \quad j \in [f_1, f_6, f_{11}]$$

It is possible that the overall network load is very light. In such cases, it is not necessary to use all three frequency channels. We define a parameter *aPeakLoadLimit*. If the total load is below this limit, only two frequency channels (preferably 1 and 11)<sup>10</sup> are used. If the peak load on any of the two channels still exceeds the *aPeakLoadLimit*, we then use all three frequency channels.

**9...1. Ignoring the Intermodulation Problem**

<sup>10</sup>Due to the susceptibility of the uplink channel 1 and 11 to downlink (6 and 11) and (1 and 6), respectively, it is desirable to avoid using channels (6 and 11) or (1 and 6) on the downlink. Therefore, when assigning two channels, the first choice is to have channels (1 and 11)

A-58

The algorithm is as follows:

- Step 1: divide 13 beams into 2 blocks. There are a total of 12 possible combinations. For each combination, the channel selection can be:  $f_1f_6, f_1f_{11}, f_6f_1, f_6f_{11}, f_{11}f_1$ , or  $f_{11}f_6$ . There are a total of 72 block-channel combinations. Assume the  $k$ th combination has the following configuration:
  - Block 1: beams 0 to  $b_k$  (0 to  $N-2$ ) are assigned to channel  $C_1$
  - Block 2: beams  $b_k+1$  to  $N-1$  (1 to  $N-1$ ) are assigned to channel  $C_2$

$$L_{C_1}^k(t) = \sum_{i=0}^{b_k} N_{iC_1}(t)$$

$$L_{C_2}^k(t) = \sum_{i=b_k+1}^{N-1} N_{iC_2}(t)$$

Then the load of channel  $C_1, C_2$  are:

Now, let the peak load on the first block for combination  $k$  be denoted

$$PL_1(k) = \max\{L_{C_1}^k(t)\} \quad t \in [0, T]$$

by:

$$PL_2(k) = \max\{L_{C_2}^k(t)\} \quad t \in [0, T]$$

then the peak load for the busiest block (channel)

$$PL_{max}(k) = \max\{PL_1(k), PL_2(k)\}$$

- is:
- Step 2: select the combination index  $R$  with the least peak load. In other words choose  $R$  such that:

$$PL_{max}(R) = \min\{PL_{max}(k)\} \quad \forall (0 \leq k \leq 71)$$

Simply described, this chooses the combination of channels and beams that minimize the peak load on any channel.

- Step 3: check that the peak load on the channel is less than aPeakLoadLimit. If not, go to three channel assignment.

### Three Channel Assignment:

- Step 1: Since the peak load on two channels exceeds the threshold, we have to assign the load to three channels. In other words we have to divide the 13 beams into 3 blocks. There are a total of 66 possible combinations. For each combination, the channel selection can be:  $f_1f_6f_{11}, f_1f_{11}f_6, f_6f_1f_{11}, f_6f_{11}f_1, f_{11}f_1f_6, f_{11}f_6f_1$ . There are a total of 396 block-channel combinations. Assume the  $k$ th combination has the following configuration:
  - Block 1: assign beams 0 to  $b_k$  (0 to  $N-3$ ) to channel  $C_1$

10

- Then the load of channel  $C_1$ ,  $C_2$ , and  $C_3$  are:

Now, let the peak load on the first block for combination  $k$  be denoted

$$PL_3(k) = \max\{L_{C_3}^k(t)\} \quad t \in [0, T]$$

then the peak load for the busiest block (channel) is:

$$PL_{max}(k) = \max\{PL_1(k), PL_2(k), PL_3(k)\}$$

- Step 2: select the combination index  $R$  with the least peak load. In other words choose  $R$  such

$$PL_{max}(R) = \min\{PL_{max}(k)\} \quad \forall (0 \leq k \leq 395)$$

that:

### 9...2. Considering the Intermodulation Problem

If considering the intermodulation problem, we would like to avoid the combinations  $f_1 f_6$  and  $f_6 f_{11}$  channels. Therefore the best approach is to avoid channel  $f_6$  all together. In other words:

The algorithm is as follows:

- Step 1: divide 13 beams into 2 blocks. There are a total of 12 possible combinations. For each combination, the channel selection can be:  $f_1 f_{11}, f_{11} f_1$ . There are a total of 24 block-channel combinations. Assume the  $k$ th combination has the following configuration:
  - Block 1: beams 0 to  $b_k$  (0 to  $N-2$ ) are assigned to channel  $C_1$

Block 2: beams  $b_k+1$  to  $N-1$  (1 to  $N-1$ ) are assigned to channel  $C_2$

Then the load of channel  $f_1, f_{11}$  is the sum of the loads of the beams assigned to those channels. In other words:

$$L_{f_1}^k(t) = \sum_{f_1} N_{f_1}(t) \quad \forall (t \in f_1)$$

$$L_{f_{11}}^k(t) = \sum_{f_{11}} N_{f_{11}}(t) \quad \forall (t \in f_{11})$$

Now, let the peak load on the first block for combination  $k$  be denoted by:

$$PL_1(k) = \max\{L_{f_1}^k(t)\} \quad t \in [0, T]$$

$$PL_2(k) = \max\{L_{f_{11}}^k(t)\} \quad t \in [0, T]$$

then the peak load for the busiest block (channel) is:

$$PL_{max}(k) = \max\{PL_1(k), PL_2(k)\}$$

- Step 2: select the combination index  $R$  with the least peak load. In other words choose  $R$  such

$$PL_{max}(R) = \min\{PL_{max}(k)\} \quad \forall (0 \leq k \leq 23)$$

that:

Simply described, this chooses the combination of channels and beams that minimizes the peak load of the busiest channel.

- Step 3: check that the peak load on the channel is less than  $aPeakLoadLimit^{11}$ . If not, go to three channel assignment.

### Three Channel Assignment:

- Step 1: Since the peak load on two channels exceeds the threshold, we have to assign the load to three channels. In other words we have to divide the 13 beams into 3 blocks. There are a total of 66 possible combinations. For each combination, the channel selection can be:  $f_1 f_6 f_{11}, f_1 f_1 f_6, f_6 f_1 f_{11}, f_6 f_1 f_1, f_1 f_1 f_6, f_1 f_6 f_1$ . There are a total of 396 block-channel combinations. Assume the  $k$ th combination has the following configuration:
  - Block 1: assign beams 0 to  $b_k$  (0 to  $N-3$ ) to channel  $C_1$
  - Block 2: assign beams  $b_k+1$  to  $p_k$  (1 to  $N-2$ ) to channel  $C_2$

---

<sup>11</sup>Considering the intermodulation problem, this parameter should be set higher to avoid the three channel combination as much as possible

A-61

## Little Joe Functional Specification

Block 3: assign beams  $p_k+1$  to  $N-1$  (2 to  $N-1$ ) to channel  $C_3$

Then the load of channel  $f_1, f_6$ , and  $f_{11}$  are:

$$\begin{aligned} L_{f_1}^k(t) &= \sum_{f_1} N_{f_1}(t) & \forall (t \in f_1) \\ L_{f_6}^k(t) &= \gamma \sum_{f_6} N_{f_6}(t) & \forall (t \in f_6) \\ L_{f_{11}}^k(t) &= \sum_{f_{11}} N_{f_{11}}(t) & \forall (t \in f_{11}) \end{aligned}$$

The parameter  $\gamma$  is the bias factor for channel 6 (*aChannelSixBiasFactor*) used to give a larger weight to channel 6 in order to penalize it's selection.

Now, let the peak load on the first block for combination  $k$  be denoted by

$$\begin{aligned} PL_1(k) &= \max\{L_{f_1}^k(t)\} & t \in [0, T] \\ PL_2(k) &= \max\{L_{f_6}^k(t)\} & t \in [0, T] \\ PL_3(k) &= \max\{L_{f_{11}}^k(t)\} & t \in [0, T] \end{aligned}$$

then the peak load for the busiest block (channel) is:

$$PL_{max}(k) = \max\{PL_1(k), PL_2(k), PL_3(k)\}$$

- Step 2: select the combination index  $R$  with the least peak load. In other words choose  $R$  such

$$PL_{max}(R) = \min\{PL_{max}(k)\} \quad \forall (0 \leq k \leq 395)$$

that:

### 1.24.2

#### 17.1 Emergency exit

If the  $i$ th traffic radio has interference larger than *aJamInterference* in the last measurement period, it has to move to the remaining channels.

## 10. Downlink Traffic-Shaping

A-62

The traffic-shaping functionality is implemented to limit the downlink load to a stable operating point and this also ensure fairness between uplink and downlink traffic load.

This section specifies the traffic shaping functionality implemented in LittleJoe. The scope is limited to specifying the functional architecture, associated algorithms and mechanisms required to implement the feature.

## 1.25

### 17.3 Traffic-Shaping Requirements

In this section we specify requirements for the traffic shaping feature. Prior to that, we define the following terms

- **Load:** the total traffic in bits/sec offered to the MAC layer (either at AP or at client) by the higher layers (e.g. TCP/IP).
- **Operating Point:** downlink load for which uplink failure rate is less than `aMaxFailureRate` (value TBD).

The requirements for the traffic shaping feature are as follows:

- Traffic shaping needs to determine the operating point for the panel and limit the downlink load to it.
- The estimation of operating point needs to be done dynamically based on changing value of uplink failure rate.

## 1.26

### 17.3 Traffic-Shaping Architecture

The traffic shaping feature is implemented at the network layer. As shown in *Functional Placement of th*, it should be placed between the backhaul and the MAC layer. All downlink traffic including traffic generated from the backhaul and internal traffic destined from one client to another client served by the panel are passed through the traffic shaper.

A-63



# Little Joe Functional Specification

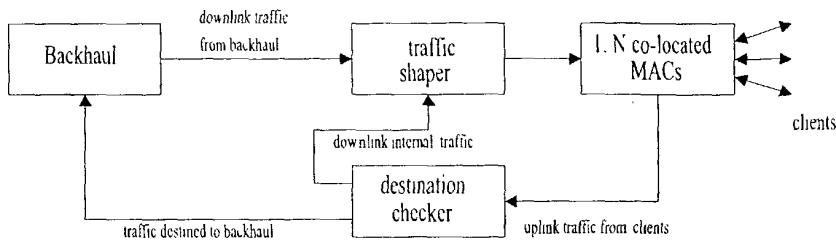


Figure 18 Functional Placement of the Traffic Shaper

Figure 19 As depicted in Architecture of Traffic Shaping Module

, the traffic shaping module consists of three components.

- Traffic queues

## **1.26.1 These queues define the granularity of traffic shaping function. A single queue is maintained for all downlink traffic. The reasons for this are discussed in**

- Gra.
- Shaper (One per queue)
  - If aggregate downlink load (offered to all radios) exceeds a threshold, system becomes unstable. The shaping function needs to estimate the load-threshold and then ensure that offered load to MAC remains below it The leaky bucket algorithm is used for traffic shaping.
- Parameter Configuration Module (One per Shaper)
  - This module is responsible for dynamically adapting shaping-parameters to meet system performance objective (i.e. keep uplink failure rate below the specified

A-64

threshold). Section 4.3 specifies operation of this entity.

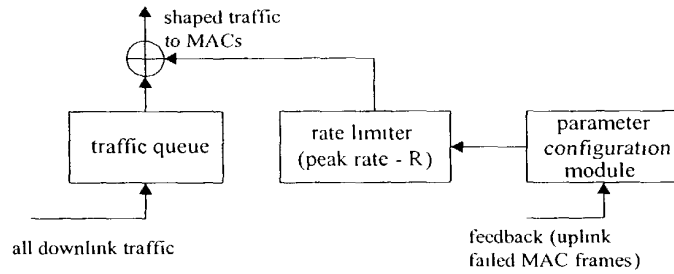


Figure 19 Architecture of Traffic Shaping Module

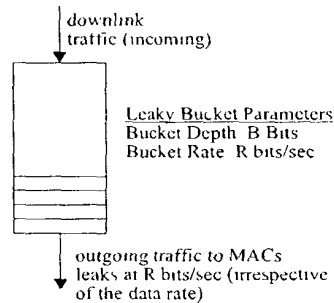
### 17.3 Traffic-Shaping Functional Description

This section specifies the functionality of traffic shaping feature. A overview of the leaky bucket algorithm and parameter update mechanism is provided. The most critical component of the specification is identifying correct values of parameters controlling leaky bucket operation. In this section, we specify the mechanisms to evaluate correct parameter values. The actual operational values are TBD.

#### 1.26.2

##### 17.1 Leaky Bucket Algorithm

The leaky bucket operation shown in *Functional Placement of th* is specified by parameters ( $R$ ,  $B$ ). The leaky bucket algorithm ensures that outgoing traffic never exceeds the specified rate  $R$ .



A-65

## Little Joe Functional Specification

Figure 20 *Leaky Bucket Operation*

Following is a description of leaky bucket operation .

- Traffic to be shaped (from the backhaul and internal client-to-client traffic) is called the incoming traffic and the shaped traffic (going to MACs) is called the outgoing traffic.
- $R$  ( $\alpha\text{MaxDownlinkLoad}$ ) is specified over a time-period  $\Delta T$  i.e. the output from leaky bucket during  $\Delta T$  should not exceed  $R$  bits.
- If during  $\Delta T$  ( $\alpha\text{ShaperWindowPeriod}$ ), the incoming traffic is less than  $R$  bits then all traffic is allowed to pass through the bucket.
- If incoming traffic size is greater than  $R$  bits, then up to  $R$  bits are transmitted and remaining packets –called violating traffic – are queued (In some implementations, violating traffic can be dropped)
- In the next  $\Delta T$  duration, the shaping criterion is applied to queued packets and newly arrived packets, i.e. up to  $R$  bits are transmitted.
- If size of queued packets exceeds the bucket depth,  $B$  ( $\alpha\text{MaxQueueSize}$ ) bits, then further incoming packets are dropped.

### 1.26.3

#### 17.1 Granularity of Operation

Traffic shaping can be defined for either single queue or there can be multiple queues (e.g. per client queue or per beam queue). Since the combined load to all AP's in the panel is the critical factor in uplink performance, it is recommended to maintain a single queue as input to the traffic shaper.

Besides if multiple queues are maintained and load thresholds are defined for each queue, it may results in unused capacity in some scenarios.

### 1.26.4

#### 17.1 Dynamic Parameter Update

The operating point for LJ system changes depending on traffic characteristics at any given time, e.g. as uplink load increases the operating point shifts to lower downlink loads. Thus it is required to continuously measure certain metrics and determine the optimal value of the operating point.

The dynamic parameter update entity is responsible for implementing this functionality. It defines only one traffic metric.

66

A-66

#### 10...1. Traffic Metric

Following metric is measured.

- Uplink failed frames (*aNumUplinkFailedFrames*)
  - This metric is indication of number of uplink frames lost at AP(s).
  - It includes retransmitted (and corrupted) frames in uplink.
  - It is not a direct indication of failure rate seen by clients (because a client could succeed in transmitting a frame after  $N$  retries, where  $N < \text{Max-Retry-Limit}$ ).
  - It is an indication of the extent of hidden beam problem.

The number of failed frames should be less than *aMaxFailureRate* during the measurement window (*aShapingMeasurementPeriod*).

The measurement window (*aShapingMeasurementPeriod*) for this metric is the duration for which the traffic metric is measured to determine whether operating point needs to change.

The parameter update module entity also implements the operating point estimation algorithm, which is specified in the next section.

#### 1.26.5

#### 17.1 Operating Point Estimation Algorithm

The algorithm is as follows.

- Select an initial value for the operating point (a table, generated from simulations provides the operating points for different system configurations, e.g. with and without CCA/CBF). This value is the *aMaxDownlinkLoad* ( $R$ ) of the leaky bucket.
- Monitor the uplink failure rate (*aNumUplinkFailedFrames*) during measurement window. If the failure rate is higher than the specified threshold (*aMaxFailureRate*)
  - Reduce operating point by  $\Delta d$  (*aShapingDecrementStep*) (i.e. reduce the value of  $R$  by  $\Delta d\%$ ).
  - Measure *aNumUplinkFailedFrames* during *aShapingMeasurementPeriod*.

While *aNumUplinkFailedFrames* is higher than *aMaxFailureRate*:

- Reduce operating point exponentially (i.e. reduce operating point by  $2\Delta d\%$  in 2<sup>nd</sup> pass,  $4\Delta d\%$  in 3<sup>rd</sup> pass and so on, where a pass is defined as *aShapingMeasurementPeriod*).
- Repeat this process for *aMaxDecrementCount* ( $N$ ) times i.e. if  $N = 4$ , then maximum reduction in operating point will be  $8\Delta d\%$  and operating point will

A.67

#### Little Joe Functional Specification

remain fixed at that value while *aNumUplinkFailedFrames* is higher than *aMaxFailureRate*.

If *aNumUplinkFailedFrames* drops below *aMaxFailureRate*

- Increase operating point by  $\Delta t\%$  (*aShapingIncrementStep*)
- Measure *aNumUplinkFailedFrames* during *aShapingMeasurementPeriod*.
- While *aNumUplinkFailedFrames* remains below *aMaxFailureRate*
  - Increase operating point exponentially i.e. by  $2\Delta t\%$  in 2<sup>nd</sup> pass,  $4\Delta t\%$  in 3<sup>rd</sup> pass and so on.

### 1.27

#### 17.3 Provisioned and Internal Parameters

This section lists the parameters which control the operation of the traffic shaping feature.

The only parameter provisioned by the network management element is:

- *aMaxFailureRate*: Uplink Failure Rate Threshold ( $F\%$ ) – Provisioned (Typical value is between 2-5%)

The internal parameters are:

- *aMaxQueueSize*: Leaky bucket depth  $B$  (bits)
- *aShaperWindowPeriod*: Leaky bucket time-period  $\Delta T$  – (Typical Value is 1 second)
- *aShapingIncrementStep*: Operating point exponential increment step size ( $\Delta t\%$ ) - Provisioned
- *aShapingDecrementStep*: Operating point exponential decrement step size ( $\Delta d\%$ ) – Provisioned
- *aMaxDecrementCount*: Maximum number of passes for exponential reduction of operating point ( $N$ ) - Provisioned
- *aShapingMeasurementPeriod*: Measurement Window duration – Provisioned
- *aMaxDownlinkLoad*: Leaky bucket rate  $R$  (bits/sec) – Dynamically updated depending on estimated operating point
- *aNumUplinkFailedFrames*: Number of uplink failed frames in the current measurement period (*aShapingMeasurementPeriod*)<sup>12</sup>

A-68

Case 2:23-cv-00202-JRG-RSP Document 86-5 Filed 08/12/24 Page 78 of 156 PageID #: 2512

## 11. The Scanning Radio

Figure 4 A scanning receiver in "promiscuous mode" and a "beam" antenna switch is used to obtain signal strength and interference information from stations on different beams as shown in *The scanning radio*

The scanning radio has two states scan mode and roaming mode. The state machine for the scanning radio is shown in *The state machine for the scanning radio*.

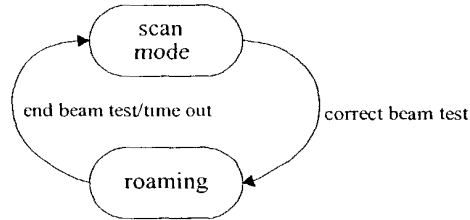


Figure 21 *The state machine for the scanning radio*

### 17.3 Scan Mode

While in the scan mode, the radio periodically scan the 13 beams on the 3 channels and collects activity information and fills the appropriate tables (39 columns). The table has the following contents described in detail in

Channel Assignment Metrics

Channel Utilization Factor (CUF)

- Self Interference Metric (SIM)
- Overlapping Subnet Interference (OSI)
- Unidentified Interference Metric (UIM)

The table contents will be a running average of the above metrics. The radio should scan each combination for at least one second in a one-minute period. However, the scanning process may be interrupted by the roaming function.

A-69

Case 2:23-cv-00202-JRG-RSP Document 86-5 Filed 08/12/24 Page 79 of 156 PageID #: 2513

Little Joe Functional Specification

## 1.28

### 17.3 Roaming

In this mode, the scanning radio is placed alternately on the neighbouring beams of the radio under test. The radio will collect the RSSI value for each frame received from the client on the two neighbouring beams. Once, the radio has received at least one frame from each neighbouring beam, it can go back to scan mode.

If there are no frames received on both neighbouring beams for a period longer than *RoamingScanTimeout*, the radio goes back to scan mode.

Additionally, we would like to ensure that, on average, no more than half of the scanning receiver's time is spent in the roaming state. The timeout in the roaming mode (*RoamingScanTimeout*), is set to  $\min(60, T/2)$  seconds where  $T$  is the average time between requests for roaming-test measurements. The Algorithm to determine the timeout ( $T$ ) is described by the pseudo code below. The algorithm uses the function *time()* (which returns the current time in seconds) and two static variables:  $T$  and *last\_test\_time*.

The pseudo-code for the initialization of *last\_test\_time* is:

```
last_test_time = 0;
```

The pseudo-code for the processing required at the start of each roaming-test measurement is:

```
// update T and last_test_time
if (last_test_time == 0) then
// initialize T
T = 60;
else
// update T
T = 0.9 * T + 0.1 * (time() - last_test_time);
endif
last_test_time = time();
// set the timeout
set_test_timeout(min(60seconds, T/2));
```

## 1.29

### 17.3 State Transitions

The rules governing the state transmissions are.

70

A-70

**correct beam test:** one of the radios is being tested for the correct beam test as explained in

- Intr

**end beam test:** a decision is made on the correct beam or the timeout period `aRoamingScanTimeout` discussed in

- Intr expires.

## 12. References

- [1] M. Brewer, D. Lohman, et al "Software System Architecture Document", ViVATo Internal Document, Version 0 3, 5/9/02.
- [2] Ed Casas, "Beamforming for LittleJoe", *ViVATO Technical Report*, Feb 1, 2002
- [4] Ed Casas, LittleJoe Link Budget, *ViVATO Technical Report*, Feb 25, 2002
- [5] Siavash Alamouti, LittleJoe Link Budget Spread Sheets, May 10, 2002.
- [6] Ed Casas, LittleJoe Beamforming, *ViVATO Technical Report*, Feb. 1, 2002
- [7] Jim Brennan, "LittleJoe Mac Model", *ViVATO Technical Report*, March 1, 2002
- [8] G. Anastasi, et al., "MAC Protocols for Wideband Wireless Local Access. Evolution Towards Wireless ATM", *IEEE Personal Communications Magazine*, Oct 1998, pp 53-64
- [9] R. Guesalla, "Characterizing the Variability of Arrival Processes with Indexes of Dispersion", *IEEE JSAC*, vol 9, no. 2, Feb 1991, pp. 203-11
- [10] W. E. Leland, et al "on the Self-Similar Nature of Ethernet Traffic," *IEEE/ACM Transactions on Networking*, vol 2, no. 1, Feb. 1994, pp 1-15
- [11] S. Deng, "Empirical Model of WWW Document Arrivals at Access Link", in the *Proceedings of ICC'96*
- [12] L. Greenstein and V. Ercog, "Gain Reductions Due to Scatter on Wireless Paths with Directional Antennas," *IEEE Communication Letters*, vol 3, no. 6, June 1999, pp 169-171
- [13] M. J. Gans, R. A. Valenzuela, J. H. Winters, and M. J. Carloni, "High Data Rate Indoor Wireless Communications Using Antenna Arrays," in *Proceeding of 6th International Symposium on Personal, Indoor and Mobile Radio Communications*, vol 3, pp 1040-1046, 1995
- [14] G. German, Q. Spencer, L. Swindlehurst, and R. Valenzuela, "Wireless Indoor Channel Modeling: Statistical Agreement of Ray Tracing Simulations and Channel Sounding Measurements," in *Proceeding of 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol 4, pp 2501-2504, 2001

A-71



Little Joe Functional Specification

- [15] J. G. Wang, A. Mohan, and T. Aubrey, "Angles-of-Arrival of Multipath Signals in Indoor Environments," in *Proceeding of Vehicular technology Conference*, vol. 1, pp 155-159, 1996.
- [16] W.C. Jakes, ed., *Microwave Mobile Communications* Wiley, 1974
- [17] J. Kivinen, X. Zhao, and P. Vainikainen, "Empirical Characterization of Wideband Indoor radio Channel at 5.3 GHz," *IEEE Transactions on Antennas and Propagation*, vol. 49, no. 8, Aug. 2001, pp. 1192-1203
- [18] J. Medbo and J. E. Berg, "Simple and Accurate Path Loss Modeling at 5 GHz in Indoor Environments with Corridors," in *Proceeding of the 52nd Vehicular technology Conference*, vol. 1, pp 30-36, 2000.
- [19] H. Hashemi, "The Indoor Propagation Channel," *Proceedings of the IEEE*, vol. 81, no. 7, July 1993, pp. 943-968
- [20] J.E. Berg, "Building Penetration Loss along Urban Street Microcells," in *Proceeding of PIMRC '96*, vol. 3, pp 795-797, 1996
- [21] E. Damosso and L. Correia, eds, *COST 213 Final Report - Digital Mobile Radio - Towards future Generation Systems* European Commission, Directorate General XIII, 1999. Report Number EUR 18957 (ISBN 92-828-5416-7)
- [22] J. D. Kraus, *Antennas*. McGraw-Hill, 1950.
- [23] J. E. Hudson, *Adaptive Array Principles*. Peter Peregrinus and IEE, 1981.
- [24] A. V. Oppenheim and R. W. Schaffer, *Digital Signal Processing*. Prentice-Hall, 1975.

[REDACTED]

A-72

Little Joe Functional Specification

## Part 2: Software System Architecture

### 13. Introduction

The purpose of this document is to describe at a high level the software features of Little Joe and how those features map onto system tasks. The types of interaction between tasks are also described.

Little Joe is based on the Linux operating system. The kernel has been customized such that only devices physically present on our system are supported. The system will contain the following components:

- PowerPC processor (integrated with DMA, Memory Controller and PCI Bus)
- SODIMM memory (64-128Mbytes)
- Onboard FLASH (32-64Mbytes)
- Three Ethernet controllers (10/100)
  - Secure management
  - Back haul
  - Daisy chain to the next panel
- The "King Arthur" custom PCI bridge
  - Connects to 11 802.11b chipsets
- Two RS232 ports
  - Management Console
  - Debug
- Temperature sensor
- Interface into the "Merlin" beam steer component
- PCI Interface to an 802.11b Search MAC

Software other than Linux that is used is:

**PPCBoot**

<http://ppcboot.sourceforge.net>

**PCMCIA**

<http://pcmcia-cs.sourceforge.net>

**Wireless Tools**

[http://www.hpl.hp.com/personal/Jean\\_Tournihes/Linux/Tools.html](http://www.hpl.hp.com/personal/Jean_Tournihes/Linux/Tools.html)

**CISH**

<http://www.tarball.net/cish/>

**Apache**

<http://www.apache.org>

**SNMP**

<http://www.net-snmp.org>

A-73

- GNU C version 2.95.3
- GNU gLibC version 2.2.4
- CVS

A-74

Little Joe Functional Specification

## 14. Software Overview

The software leverages a reasonable amount of IP from the open source community ranging from the boot code to the protocol stacks themselves. The Mabuhay software IP (for the 1<sup>st</sup> release) resides in the following components:

- The integration of the Linux components onto the custom PCB.
- The King Arthur driver code.
- Merlin beam steering algorithms and its driver.
- Access Point control software.

The rest of the code is primarily open source and the job is largely an integration effort to get all of the components to operate in our target system in a manner which is compatible with the product requirements.

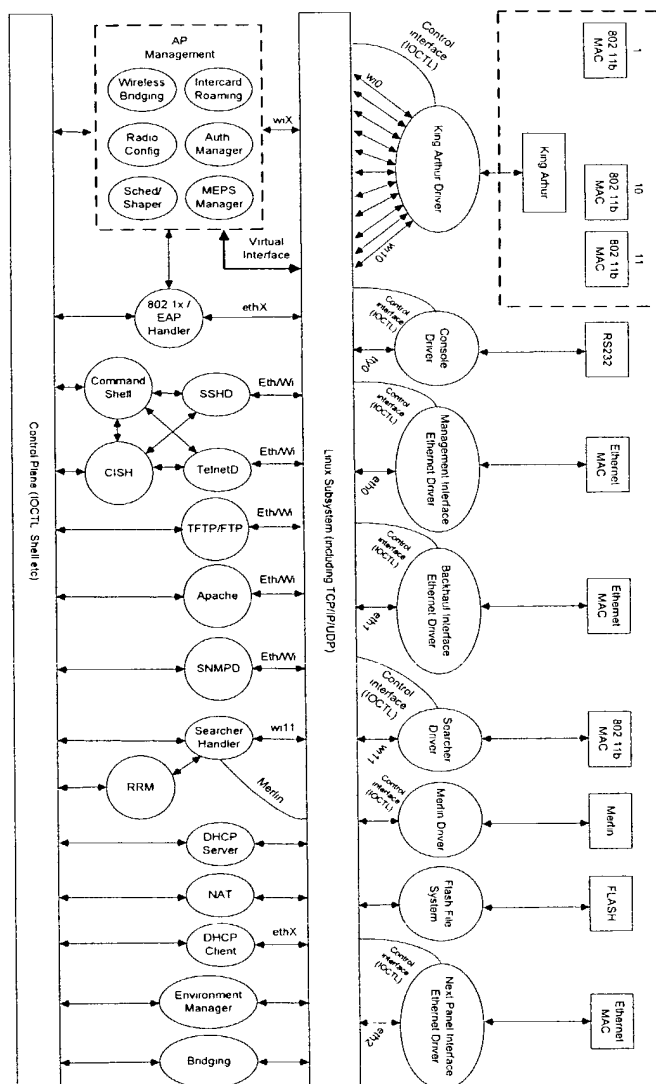
## 17.3 Software not covered by this Document

This document covers the modules and subsystems which will be present when the system is running. However, there are other important components which are not covered. These are:

- PowerPC-boot
  - Diagnostics
    - FLASH
    - RAM
    - Ethernet interfaces
    - Serial interfaces
    - Temperature calibration
    - Fan controls
  - Flash file system for Kernel
- Linux Kernel Configuration
  - Providing the necessary drivers for essential components

A-75

## 17.3 Software Module Overview



A-76

Case 2:23-cv-00202-JRG-RSP Document 86-5 Filed 08/12/24 Page 86 of 156 PageID #: 2520

## Little Joe Functional Specification

**15. Control Plane**

To simplify the connectivity between modules the diagram uses a control plane which is the lingo for any one or many of the following inter-process communication techniques:

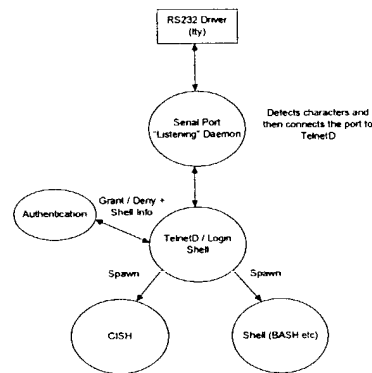
- RPC
- A shell executing a script and extracting output (textual or otherwise)
- IOCTL
- Protocol stack interface (socket or otherwise)
- Shared memory
- File

On a module by module basis a developer will decide and publish the interface to their code. The code-base is using open source software, so if the module is expected to contain valuable intellectual property that is proprietary to Mabuhay then the interface should be disjoint – for example the beam steering software should not be code compiled into the CISH command line interpreter.

**16. Drivers****17.3 Console Driver**

The management console interface is simply and RS232 port configured, out of the box, to be 9600 baud, 8 bits, no stop bits.

The manner in which this operates when a user connects a terminal to it should be as follows:



A-77

### 17.3 Ethernet Drivers

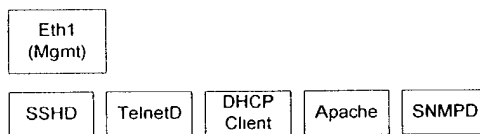
There are three Ethernet interfaces on the product. The interfaces plumb into the underlying Linux subsystem in an identical manner. However, they differ in the services offered. The following diagrams show which daemons are listening to which ports.

#### 1.29.1

##### 17.1 Secure Management

This port is intended as a separate “secure” out-of-band port which may be wired into an existing infrastructure to manage the device. The port is secure, not because it is encrypted (it is not), but because it is physically separate and will not participate in packet forwarding. Under no circumstances can data packets sent into the wired or wireless interfaces be forwarded to this port. Device management over this interface may be performed by CLI, HTTP(S) and SNMP

Therefore the stack is as follows:



A driver should be developed to interface the target device to the Linux subsystem in a standard (ethX) manner. If no such driver exists for the chosen part then one will be developed and offered up to the open source community, since the driver contains no Mabuhay proprietary intellectual property.

Configuration settings of this interface should be as follows:

- The ability to set the interface to Auto-negotiate
  - The interface should advertise and be able to negotiate to 10 Mbps, 100 Mbps, Full Duplex, Half Duplex, no flow control.
- The ability to set the interface manually to:
  - 10 Mbps, Half Duplex
  - 100 Mbps, Half Duplex
  - 10 Mbps, Full Duplex
  - 100 Mbps, Full Duplex
- Out of the box:
  - Auto-negotiate enabled

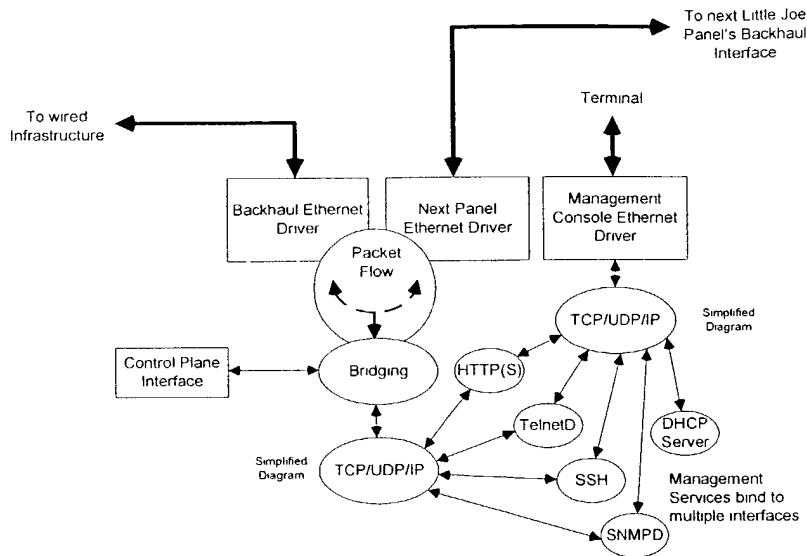
#### 1.29.2

A-78

## Little Joe Functional Specification

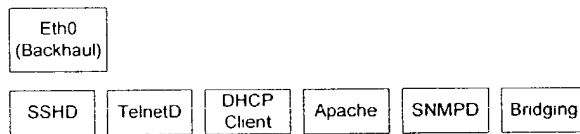
**17.1 Backhaul / Daisy-chain for the Next Panel**

This interface provides the connection from Little Joe to the wired infrastructure. Packets which may transfer into and out from this interface are as follows:



The diagram shows that management of the unit may be performed via the dedicated management port or the wired interfaces. However, via the control plane, management may be disabled to any interface for specific services. For example, each management system will have a list of "allowable" interfaces that it may serve itself to. Out of the box, CISH (which is spawned by TelnetD upon authentication clearance) may be disabled for certain interfaces, such as the wireless interface (wl3), wired backhaul, or next panel.

A typical setup of allowable services for the Backhaul interfaces may be:



The DHCP Client Service is shown. This is because a normal operating characteristic of existing access points from factory power-on is for it to gather its own IP address and perform bridging between the wired, wireless and, in the case of Little Joe, the next panel interface.

A-79



**1.30**

### **17.3 Source for Wireless Drivers**

Many of the wireless cards, such as those from Intersil and Agere are supported by source available in the open-source community. For the Intersil cards the following are available:

<http://people.ssh.com/jkm/Prism2/>

This is the so-called Host-AP mode driver sponsored by Intersil that allows a client card to become an access point, except that management functions such as association and authentication are handled by the CPU, but Beacon Generation and Probe Responses are performed in hardware.

<http://www.linux-wlan.com/linux-wlan/>

The WLAN-driver project is another driver for the same card, except this is for client-mode applications.

Both projects utilize the iwconfig utilities which use an IOCTL interface to talk to the drivers.

The drivers live in kernel space and can be bound in with the kernel or loaded at runtime.

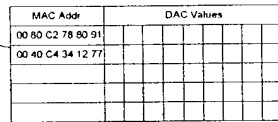
Drivers for wireless interfaces for Linux have been supplied in just object code, such as those from Cisco and Nokia, and also in source form, from the likes of Intersil and Agere.

**1.31**

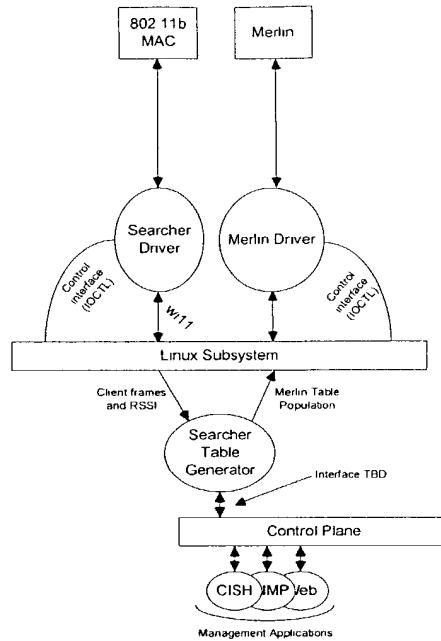
### **17.3 Searcher and Merlin Interfaces and Functions**

The searcher MAC is designed to determine where clients are and build a table for the Merlin beam steerer. In order to understand the control system behind these two functions, consider the following diagram which shows how each of the wireless APs in the system interface to Merlin:

**A-80**



The sector switch allows the CPU to change the receive beam that the searcher MAC is focused on. By a process of sampling each of the sectors a map can be built of client MAC addresses and their respective receive strength signal indications (RSSI). By using simple math it is then possible to determine their location. The location consists of a series of values which can be fed into a vector modulator in order to “shine” a beam onto a specific client. These values determine both the reach (power) and the direction. Thus, when an AP wishes to transmit a frame to a client, the Merlin logic is able to see which MAC address it to be communicated with; adjust the beam to the appropriate location, and transmit the frame. The software processes which make this happen are as follows:



### 17.3 Scheduler / Shaper

Due to the nature of 802.11 networks when combined with directional antennas, it is important to be able to identify situations which would or could adversely affect performance. For example, it is possible that one user may wish to perform a large download, and due to this it could, when taken to an extreme, cause other clients to be starved of service. This module identifies when such situation occur and performs the necessary packet shaping (bandwidth management) to ensure that the unit may still be able to see when another client is requesting service. There are numerous ways of achieving this and the exact algorithm is still to be determined. However, the positioning of the module in the system is clear – it must be between the back and wireless devices and be able to set up queues and bandwidth limits on a per client basis. Therefore, within the architecture it would live as follows for a typical packet flow:

A-82



### 17.3 FLASH

This driver needs to present an interface to Linux which is compatible with a regular block device such as a hard drive. However, since Flash has a limited number of write cycles, and the number of parameters that can be changed in the system is significant and likely to get a lot larger as the product develops, changes in configuration will need to be committed through a user "save" command. Such an action will write the configuration to Flash rather than individual writes for each small change a user makes.

Note: Research is continuing into a MTD Driver and JFFS Journaling Flash File System Drivers.

Component	Filesize	Memory
Httpd	550K	3000K
Httpd.conf	50K	
libperl.so	1.2 MB	
Cish	150K	500K
Net-snmp	20K	
Total	~2MB	3500K

This table shows initial estimations of management module resource usage. Need clarification of data items in the table. Some of these items are daemons, running all of the time. Others are one instance per invocation. Cish is dependent on lots of other elements, such as ipchains, brctl and ifconfig.

Deleted: ¶

## 2. Management Interfaces

There are three core management interfaces.

- CLI (RS232, Telnet, SSH, ASCII File)
- HTTP
- SNMP

All three management interfaces are available through the RS232 console port, Backhaul (eth0), Management (eth1), Next Panel (eth2) and the Wireless interfaces. Instances of the HTTP, SNMP, Telnet/SSH daemons will be bound to the eth0, eth1, eth2 and wix interfaces. The user will have the ability to disable management services on individual or all interfaces.

The following list is an initial draft of the features in the Little Joe AP which will be configured and monitored using the CLI, HTTP, and SNMP management services.

- 1) Basic Setup
  - a) SSID
  - b) IP
- 2) Chassis
  - a) Slots
  - b) Ethernet Ports

Formatted: Bullets and Numbering

14-84

Little Joe Functional Specification

- c) Antenna
  - i) Calibration
- d) Radios
- e) Environment
- 3) Layer 2
  - a) Ethernet
  - b) 802.11
  - c) Bridging
  - d) VLAN
  - e) Filters
  - f) Statistics
- 4) Layer 3
  - a) Basic IP Configuration
  - b) SSH
  - c) HTTP
  - d) Telnet
  - e) SNMP
    - i) MIB 1 & 2
    - ii) Mabuhay MIBs
    - iii) Dot1 and dot11 MIBs
    - iv) Traps
  - f) DHCP
  - g) NAT
  - h) NTP
  - i) FTP
  - j) Filters
  - k) Statistics
- 5) AP Manager
  - a) Associations
  - b) Local AP Roaming
  - c) Inter AP Protocol / Wireless Distribution System
  - d) Multi-AP Configuration (4 panel setup)
- 6) Security
  - a) WEP
    - i) 40-bit
    - ii) 128-bit
  - b) Authentication
    - i) Open
    - ii) Shared
    - iii) EAP
    - iv) Radius
- 7) Utilities
  - a) Users/Administrators
  - b) Preferences
  - c) Searching and Sorting

A-85

- d) Configuration File
  - i) Import/Export
  - ii) Version(s)
- e) Firmware
  - i) Update
  - ii) Version(s)
- f) File
- g) Syslog
- h) TFTP
- i) FTP
- 8) Diagnostics
  - a) Events/Logs/Traps
  - b) Network
    - i) Ping
    - ii) Traceroute
  - c) Radio
  - d) Antenna
    - i) Calibration
  - e) Merlin
    - i) Dump Beam Coefficient Data Table

### **17.1 HTTP**

The HTTP(S) service provides a web based management interface. The complete set of CLI commands will be available through the Web screens. Access to the web interface is controlled by the management authentication process. The web interface will consist of standard HTML and CGI scripts. We are evaluating the mod\_perl Apache module for an additional server-side scripting tool. The goal is to support Internet Explorer, Netscape, Mozilla and other browsers on multiple platforms. Apache is well supported and is a solid base for future features including XML, SOAP, JSP, ASP and Web Services.

The HTTPD can be configured to bind on the Backhaul (eth0), Management (eth1), Next Panel (eth2) and Wireless interfaces (wix). The HTTP service can be configured to bind on all or none of the interfaces.

The Apache HTTP server with SSL/TLS libraries provides a secure HTTPS platform.

#### **User Interface (UI)**

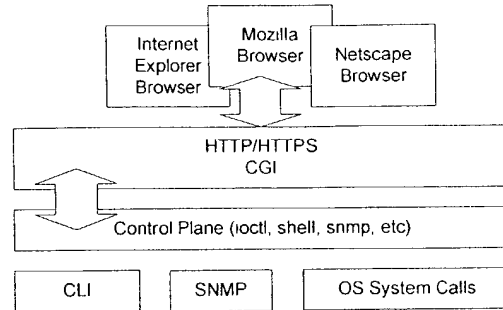
- Forms
- Help

The forms provide standard Web UI features including edit, list boxes, radio buttons, and tables. The UI advances the CLI giving the user for example a single page to do basic setup instead of a series of commands. Also, wizards (a series of forms) will be developed to walk the user through complex configuration tasks. Advanced UI features – Graphic view of the chassis, Network Topology, Graphical views of beam status will be considered in the future.

**A-86**

08/12/24 11:04:12

## Little Joe Functional Specification

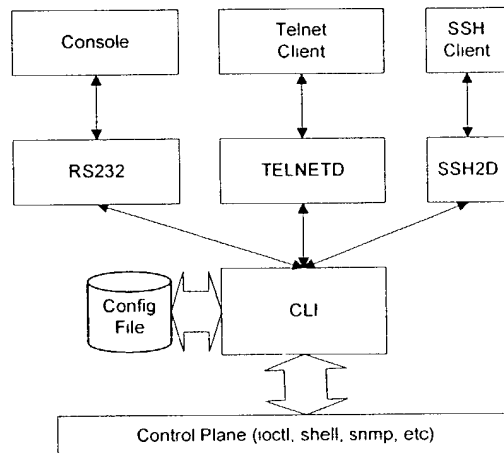


This diagram shows the high-level HTTP module interfaces.

### 1.32

#### 17.3 CISH (CLI)

CISH is a command line (CLI) interface based using a simple Cisco-like interface, including a simple help system and command completion and command history. Command syntax is defined in tables, making it fairly easy for developers to add commands. CISH validates the command parameters, then will use the control plane interface to configure and get status information. Configuration is stored in a ASCII text file. User authentication is password only (no user ID). An ASCII Text File provides a persistent store of the configuration. The file contains the CLI commands used to configure the device. The file can be used to configure multiple devices with a common setup.



This diagram shows the high-level CLI interfaces.

A-87



### 17.3 User Manager

Provide a facility to add/remove users for management of the device. Users will have associated capabilities. For example, an operator may have read-only access, and a admin user have full read-write capabilities.

Deleted: user

Deleted: n

### 17.4 SSHD

Secure Shell from [www.openssh.org](http://www.openssh.org) – contains support for SSH1 and SSH2. Secure Shell provides encryption, authentication and tunneling capabilities. The OpenSSH package includes the ssh program which replaces telnet and rlogin, scp replaces rcp and sftp which replaces ftp.

OpenSSH does not support any patented transport algorithms. In SSH1 mode, only 3DES and Blowfish are available options. In SSH2 mode, only 3DES, Blowfish, CAST128, Arcfour and AES can be selected. The patented IDEA algorithm is not supported

OpenSSH provides support for both SSH1 and SSH2 protocols.

Since the RSA patent has expired, there are no restrictions on the use of RSA algorithm using software.

### 17.5 Telnet

The Telnet daemon module will spawn a CISH shell. The telnetd service can be bound to selected interfaces. The telnetd service can be enabled/disabled.

### 1.33

### 17.6 SNMPD

The SNMP agent will be based on the NET-SNMP open source project formerly known as the UCD-SNMP package, originally based on the Carnegie Mellon University SNMP implementation (version 2.1.2.1).

The net-snmp package supports SNMPv1, SNMPv2 and SNMPv3. TDB if we need to include v3 support.

Support for MIB 1 and MIB 2 as well as Mabuhay private MIBs and Traps. Register Mabuhay private mib with iana.org.

A SNMP access function framework will provide a standard interface for SNMP set, get, getNext and Traps for other developers to integrate their modules into the SNMP agent.

## 18. Environmental Control

A-88

## Little Joe Functional Specification

**18.1 Temperature and Fan Control**

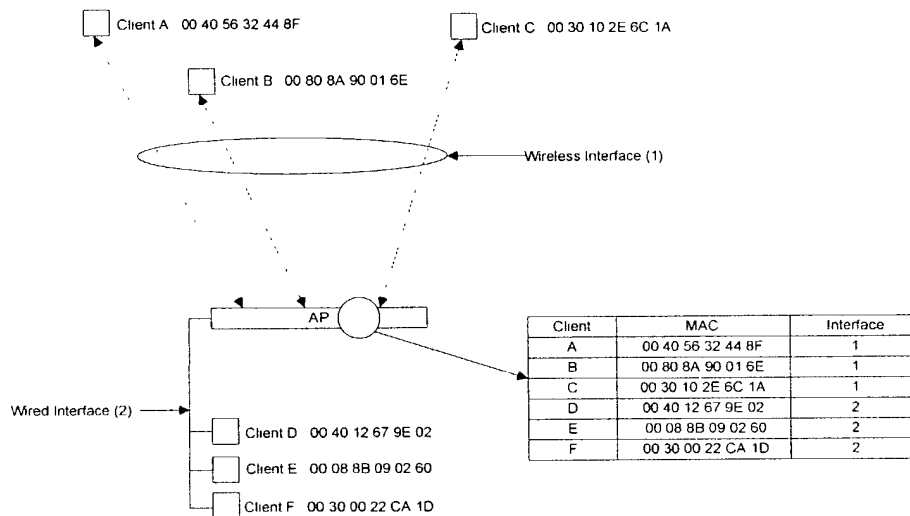
The temperature module is fairly simple – it simply needs to check the temperature against predefined limits (programmable) and take appropriate action when the system is above or below temperature (or approaching it). Changes in temperature will result in the closed loop control system operating the fans (below). However, when cooling fails, the following actions can be taken:

- Issue a TRAP to an SNMP NMS
- Write error information in the local store syslog. Optionally notify a UNIX Syslog daemon.
- Issue warnings to the console, CLI and web interface.

**19. Wireless Control****19.1 Wireless Bridging**

The function of this module will be dependent on the actual firmware that is selected for the Access Points in the system. However, if the chosen firmware does its own wireless bridging, then this section describes its function. However, should it not, then the function will be provided through the Bridging Manager described later.

From an AP perspective, associated clients would populate a location database as follows:



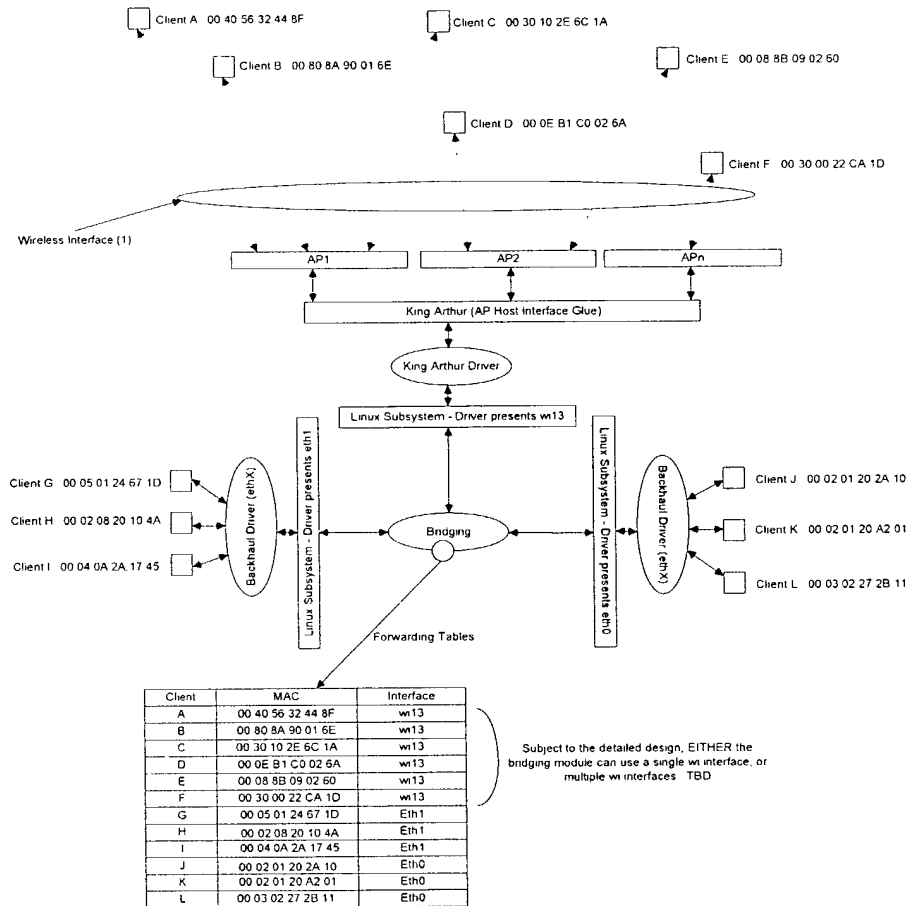
Traffic local the AP is kept local, and only traffic destined for the wired backhaul is transferred out of the AP itself.

A-89

## 1.34

**19.2 Centralized Bridging**

It is desirable to have all packets come to a central bridging manager as follows:



In this example, the bridging manager running on the host CPU has control of all of the interfaces for data forwarding. In the example the table is indexed by MAC address. However, although release 1 may not be VLAN-aware, the extensibility of the table and indexed must be considered in the detailed design.

A - 90

Little Joe Functional Specification

The diagram above also shows the wireless interfaces under a single wi13 interface. The driver for wi13 would then do an additional lookup to determine the actual wireless interface to which a user may be attached.

However, an alternative which may be concluded is a better choice (dependent on final detailed design for that module), may be to bridge each of the wireless interfaces together and provide a layer 3 IP interface in the same broadcast domain. This would remove the necessity of providing two area for layer 2 lookups to be performed

### **19.3 Radio Configuration Manager**

SSID, SSID Broadcast, Data Rates, RTS/CTS settings, Default Radio Channel, # of Wireless clients.

#### **1.35**

### **19.3 Inter-card Roaming Manager**

#### **1.36**

### **19.3 Mabuhay Enhanced Performance System (MEPS)**

#### **1.37**

### **19.3 Security**

#### **1.38**

### **19.3 Authentication Manager**

The authentication manager will be responsible for handling incoming 802.11b authentication requests. The authentication manager receives all authentication frames and determines the authentication mechanism to use based on the bits in the authentication type field. Based on the authentication type the request is handed-off to the authentication handler which will complete the authentication process

#### **1.39**

### **19.3 Authentication**

Before a client device can gain access to the AP and the network, it must be authenticated. There are four AP authentication mechanisms: Shared key, Open Authentication, MAC address, and EAP.

- Open Authentication – Allows any client to authenticate with AP, but only allows data transfer if WEP keys match.

A-91

- Shared Key – 802.11b standard but not recommended because of vulnerabilities. Sends unencrypted challenge string to the client. Client responds with encrypted response, if correct the AP allows the client to authenticate.
- MAC address – The MAC address is either checked against a local AP table of allowable MACs or sent to a RADIUS server for verification. If the MAC is not in the list of allowable MACs then the device is not authenticated.

## 1.40

### 19.3 802.1x / EAP Authentication Mechanism

802.1X defines Extensible Authentication Protocol (EAP) over LANs (EAPOL). The standard encapsulates and leverages much of EAP, which was defined for dial-up authentication with Point-to-Point Protocol in RFC 2284.

Client sends authenticate request to AP, AP sends an EAPOL-encapsulated EAP request-ID to the client. The client responds with an EAPOL-encapsulated EAP response-ID message containing the user's identification. The AP then reencapsulates this same EAP response-ID message in a RADIUS access request packet and forwards this to a RADIUS server. EAP messages are relayed between the client and RADIUS by the AP, on the client side encapsulated in EAPOL, and on the server side inside a RADIUS packet.

In the final step, the RADIUS server responds with a RADIUS access accept (or deny) packet containing an encapsulated EAP success (or failure), which the AP then forwards to the client. In the case of success, the port is considered opened for data traffic and the user authenticated.

When using dynamic session keys the RADIUS access accept will include session keys, which are used by the wireless access point to build, sign and encrypt an EAPOL key message.

This is sent to the client immediately following the EAP success message. With this information, both client and wireless access point can program their encryption keys dynamically, making the encryption more difficult to crack.

#### WEP (Wired Equivalent Privacy)

Enable/Disable WEP.

40-bit and 128-bit key setup.

## 2. Other Control

### 19.3 DHCP Client

This module is enabled by default and operates through the wired Backhaul interface. Unless a specific IP address is specified (which from the factory can only be done through the Management Console port) the device will boot and try and get its management IP address from a DHCP Server on the Backhaul interface.

A-92

CONFIDENTIAL

Little Joe Functional Specification

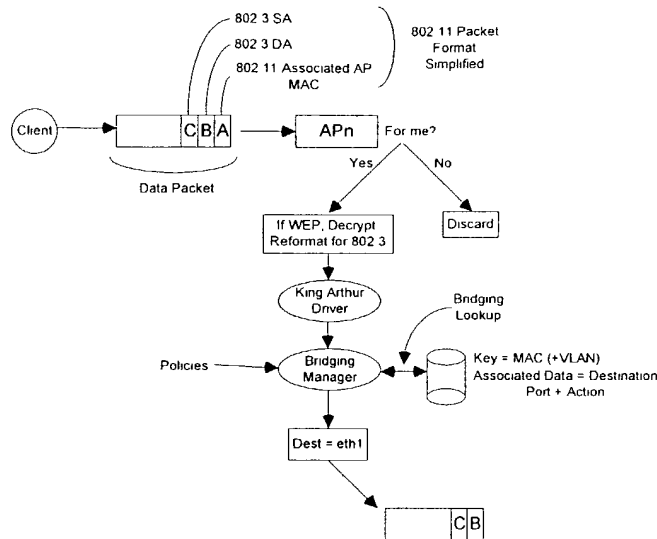
**1.41**

**19.3 DHCP Server and Network Address Translation (NAT)**

A-93

### 3. Typical Packet Walk (Bridged)

The following diagram shows the one way trip from an 802.11b client through to the 802.3 Backhaul Interface.



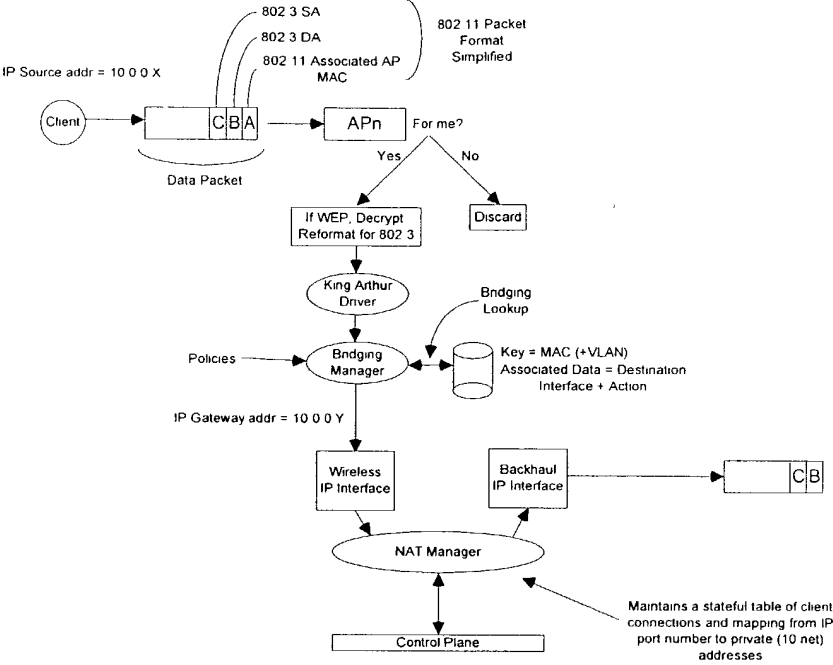
### 19. Typical Packet Walk (NAT and DHCP Server)

Another typical deployment for an access point, especially when there is no central DHCP Server, is for the access point to provide an IP gateway functionality whilst providing private (10 or 192 net) addresses for clients to communication over. The diagram below shows the modules that are utilized to get a packet from a client to a backhaul interface.

The NAT module maintains stateful information on each connection. Each session is translated such that a connection to a destination site is translated with a new source address (from a private address to the real IP address of the access point), and the source system is identified by a new IP source port number. When a frame is received from the destination on the backhaul interface a lookup is performed on the destination address port to see if it matches an existing connection. If a match occurs the NAT function will reform the packet with the private 10 or 192 net address and transmit the frame out of the appropriate wireless interface.

A-94

Little Joe Functional Specification



A-95

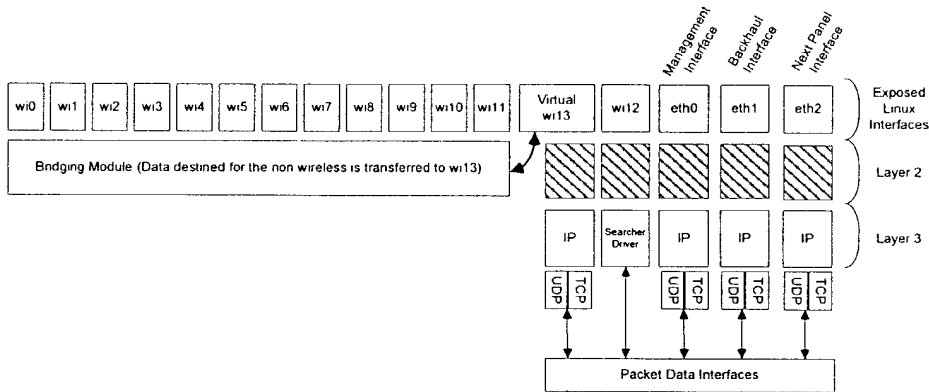


A-96

Little Joe Functional Specification

## 19. Data Packet Interfaces

The following diagram depicts the Linux interface stack along with the bindings to layer 3. The wireless interfaces present themselves to the system in much the same way as any other interface. However, the data interface through which traffic may be transmitted and received is only one of those interfaces. As discussed earlier in this design, the bridging between wired and wireless and between wireless and between wired may be the same bridging module or via two separate modules. The compliance with the Linux interface architecture allows a lot of flexibility when it comes to implementation of the featureset.



A-97

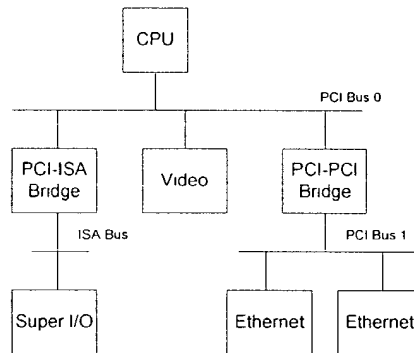
## 19. Appendix A – PCI / PCI Bridge Support in Linux

Peripheral Component Interconnect (PCI), as its name implies is a standard that describes how to connect the peripheral components of a system together in a structured and controlled way. The standard describes the way that the system components are electrically connected and the way that they should behave. This chapter looks at how the Linux kernel initializes the system's PCI buses and devices.

### 1.42.1

#### 19.4 Example PCI Based System

This is a logical diagram of an example PCI based system. The PCI buses and PCI-PCI bridges are the glue connecting the system components together; the CPU is connected to PCI bus 0, the primary PCI bus. A special PCI device, a PCI-PCI bridge connects the primary bus to the secondary PCI bus, PCI bus 1. In the jargon of the PCI specification, PCI bus 1 is described as being downstream of the PCI-PCI bridge and PCI bus 0 is up-stream of the bridge. Connected to the secondary PCI bus are the two ethernet devices for the system. Physically the bridge, secondary PCI bus and two devices would all be contained on the same combination PCI card. The PCI-ISA bridge in the system supports older, legacy ISA devices and the diagram shows a super I/O controller chip.



### 1.42.2

#### 19.4 PCI Address Spaces

The CPU and the PCI devices need to access memory that is shared between them. This memory is used by device drivers to control the PCI devices and to pass information between them. Typically the shared memory contains control and status registers for the device. These registers are used to control the device and to read its status.

The CPU's system memory could be used for this shared memory but if it were, then every time a PCI device accessed memory, the CPU would have to stall, waiting for the PCI device to

A-98

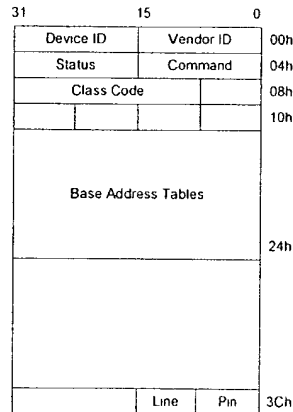
Little Joe Functional Specification

finish. Access to memory is generally limited to one system component at a time. This would slow the system down. It is also not a good idea to allow the system's peripheral devices to access main memory in an uncontrolled way. This would be very dangerous; a rogue device could make the system very unstable.

Peripheral devices have their own memory spaces. The CPU can access these spaces but access by the devices into the system's memory is very strictly controlled using DMA (Direct Memory Access) channels. ISA devices have access to two address spaces, ISA I/O (Input/Output) and ISA memory. PCI has three; PCI I/O, PCI Memory and PCI Configuration space. All of these address spaces are also accessible by the CPU with the the PCI I/O and PCI Memory address spaces being used by the device drivers and the PCI Configuration space being used by the PCI initialization code within the Linux kernel.

1.42.3

19.4 PCI Configuration Headers



Every PCI device in the system, including the PCI-PCI bridges has a configuration data structure that is somewhere in the PCI configuration address space. The PCI Configuration header allows the system to identify and control the device. Exactly where the header is in the PCI Configuration address space depends on where in the PCI topology that device is. For example, a PCI video card plugged into one PCI slot on the motherboard will have its configuration header at one location and if it is plugged into another PCI slot then its header will appear in another location in PCI Configuration memory. This does not matter, for wherever the PCI devices and bridges are the system will find and configure them using the status and configuration registers in their configuration headers.

Typically, systems are designed so that every PCI slot has it's PCI Configuration Header in an offset that is related to its slot on the board. So, for example, the first slot on the board might have its PCI Configuration at offset 0 and the second slot at offset 256 (all headers are the same length, 256 bytes) and so on. A system specific hardware mechanism is defined so that the PCI

A-99

configuration code can attempt to examine all possible PCI Configuration Headers for a given PCI bus and know which devices are present and which devices are absent simply by trying to read one of the fields in the header (usually the Vendor Identification field) and getting some sort of error. The describes one possible error message as returning 0xFFFFFFFF when attempting to read the Vendor Identification and Device Identification fields for an empty PCI slot.

#### **19.4 Layout of the 256 byte PCI configuration header**

It contains the following fields:

##### **1.42.4**

#### **19. Vendor Identification**

A unique number describing the originator of the PCI device. Interestingly, Intel's is 0x8086.

##### **1.42.5**

#### **19. Device Identification**

A unique number describing the device itself.

##### **1.42.6**

#### **19. Status**

This field gives the status of the device with the meaning of the bits of this field set by the standard. .

##### **1.42.7**

#### **19. Command**

By writing to this field the system controls the device, for example allowing the device to access PCI I/O memory,

##### **1.42.8**

#### **19. Class Code**

This identifies the type of device that this is. There are standard classes for every sort of device; video, SCSI and so on. The class code for SCSI is 0x0100.

A-100

Little Joe Functional Specification

#### 1.42.9

### 19.4 Base Address Registers

These registers are used to determine and allocate the type, amount and location of PCI I/O and PCI memory space that the device can use.

#### 1.42.10

### 19.4 Interrupt Pin

Four of the physical pins on the PCI card carry interrupts from the card to the PCI bus. The standard labels these as A, B, C and D. The Interrupt Pin field describes which of these pins this PCI device uses. Generally it is hardwired for a particular device. That is, every time the system boots, the device uses the same interrupt pin. This information allows the interrupt handling subsystem to manage interrupts from this device.

#### 1.42.11

### 19.4 Interrupt Line

The Interrupt Line field of the device's PCI Configuration header is used to pass an interrupt handle between the PCI initialization code, the device's driver and Linux's interrupt handling subsystem. The number written there is meaningless to the device driver but it allows the interrupt handler to correctly route an interrupt from the PCI device to the correct device driver's interrupt handling code within the Linux operating system.

## 19. PCI I/O and PCI Memory Addresses

These two address spaces are used by the devices to communicate with their device drivers running in the Linux kernel on the CPU. For example, some fast Ethernet devices map their internal registers into PCI I/O space. The Linux device driver then reads and writes those registers to control the device.

Until the PCI system has been set up and the device's access to these address spaces has been turned on using the Command field in the PCI Configuration header, nothing can access them. It should be noted that only the PCI configuration code reads and writes PCI configuration addresses; the Linux device drivers only read and write PCI I/O and PCI memory addresses.

#### 1.43

## 19. PCI-ISA Bridges

These bridges support legacy ISA devices by translating PCI I/O and PCI Memory space accesses into ISA I/O and ISA Memory accesses. A lot of systems now sold contain several ISA bus slots and several PCI bus slots. Over time the need for this backwards compatibility will dwindle and PCI only systems will be sold. Where in the ISA address spaces (I/O and Memory) the ISA devices of the system have their registers was fixed in the dim mists of time by the early Intel 8080 based PCs. Even a \$5000 Alpha AXP based computer systems will have its ISA

A-101

floppy controller at the same place in ISA I/O space as the first IBM PC. The PCI specification copes with this by reserving the lower regions of the PCI I/O and PCI Memory address spaces for use by the ISA peripherals in the system and using a single PCI-ISA bridge to translate any PCI memory accesses to those regions into ISA accesses.

**19. PCI-PCI Bridges**

PCI-PCI bridges are special PCI devices that glue the PCI buses of the system together. Simple systems have a single PCI bus but there is an electrical limit on the number of PCI devices that a single PCI bus can support. Using PCI-PCI bridges to add more PCI buses allows the system to support many more PCI devices. This is particularly important for a high performance server. Of course, Linux fully supports the use of PCI-PCI bridges.

**1.44**

**19. PCI-PCI Bridges: PCI I/O and PCI Memory Windows**

PCI-PCI bridges only pass a subset of PCI I/O and PCI memory read and write requests downstream. For example, in the diagram at the beginning of the appendix, the PCI-PCI bridge will only pass read and write addresses from PCI bus 0 to PCI bus 1 if they are for PCI I/O or PCI memory addresses owned by either of the Ethernet devices; all other PCI I/O and memory addresses are ignored. This filtering stops addresses propagating needlessly throughout the system. To do this, the PCI-PCI bridges must be programmed with a base and limit for PCI I/O and PCI Memory space access that they have to pass from their primary bus onto their secondary bus. Once the PCI-PCI Bridges in a system have been configured then so long as the Linux device drivers only access PCI I/O and PCI Memory space via these windows, the PCI-PCI Bridges are invisible. This is an important feature that makes life easier for Linux PCI device driver writers. However, it also makes PCI-PCI bridges somewhat tricky for Linux to configure.

**1.45**

**19. PCI-PCI Bridges - PCI Configuration Cycles and PCI Bus Numbering**

**1.45.1 Type 0 PCI Configuration Cycle Figure:**

31	11	10	8	7	2	1	0
Device Select			Func	Register	0	0	

A-102

## Little Joe Functional Specification

## 1.45.2

## 1.45.3 Type 1 PCI Configuration Cycle Figure:

31	24	23	16	15	11	10	8	7	2	1	0
Reserved				Bus		Device		Func	Register		0
											0

So that the CPU's PCI initialization code can address devices that are not on the main PCI bus, there has to be a mechanism that allows bridges to decide whether or not to pass Configuration cycles from their primary interface to their secondary interface. A cycle is just an address as it appears on the PCI bus. The PCI specification defines two formats for the PCI Configuration addresses; Type 0 and Type 1; these are shown in the figures above. Type 0 PCI Configuration cycles do not contain a bus number and these are interpreted by all devices as being for PCI configuration addresses on this PCI bus. Bits 31:11 of the Type 0 configuration cycles are treated as the device select field. One way to design a system is to have each bit select a different device. In this case bit 11 would select the PCI device in slot 0, bit 12 would select the PCI device in slot 1 and so on. Another way is to write the device's slot number directly into bits 31:11. Which mechanism is used in a system depends on the system's PCI memory controller.

Type 1 PCI Configuration cycles contain a PCI bus number and this type of configuration cycle is ignored by all PCI devices except the PCI-PCI bridges. All of the PCI-PCI Bridges seeing Type 1 configuration cycles may choose to pass them to the PCI buses downstream of themselves. Whether the PCI-PCI Bridge ignores the Type 1 configuration cycle or passes it onto the downstream PCI bus depends on how the PCI-PCI Bridge has been configured. Every PCI-PCI bridge has a primary bus interface number and a secondary bus interface number. The primary bus interface being the one nearest the CPU and the secondary bus interface being the one furthest away. Each PCI-PCI Bridge also has a subordinate bus number and this is the maximum bus number of all the PCI buses that are bridged beyond the secondary bus interface. Or to put it another way, the subordinate bus number is the highest numbered PCI bus downstream of the PCI-PCI bridge. When the PCI-PCI bridge sees a Type 1 PCI configuration cycle it does one of the following things:

1. Ignore it if the bus number specified is not in between the bridge's secondary bus number and subordinate bus number (inclusive)
2. Convert it to a Type 0 configuration command if the bus number specified matches the secondary bus number of the bridge
3. Pass it onto the secondary bus interface unchanged if the bus number specified is greater than the secondary bus number and less than or equal to the subordinate bus number.

So, if we want to address Device 1 on bus 3 of the topology defined in the section entitled "PCI-PCI Bridging Step 3" we must generate a Type 1 Configuration command from the CPU. Bridge1 passes this unchanged onto Bus 1. Bridge2 ignores it but Bridge3 converts it into a Type 0 Configuration command and sends it out on Bus 3 where Device 1 responds to it.

A-103



It is up to each individual operating system to allocate bus numbers during PCI configuration but whatever the numbering scheme used the following statement must be true for all of the PCI-PCI bridges in the system:

"All PCI buses located behind a PCI-PCI bridge must reside between the secondary bus number and the subordinate bus number (inclusive) "

If this rule is broken then the PCI-PCI Bridges will not pass and translate Type 1 PCI configuration cycles correctly and the system will fail to find and initialize the PCI devices in the system. To achieve this numbering scheme, Linux configures these special devices in a particular order. Section Assigning PCI Bus Number describes Linux's PCI bridge and bus numbering scheme in detail together with a worked example.

## 1.46

### 19. Linux PCI Initialization

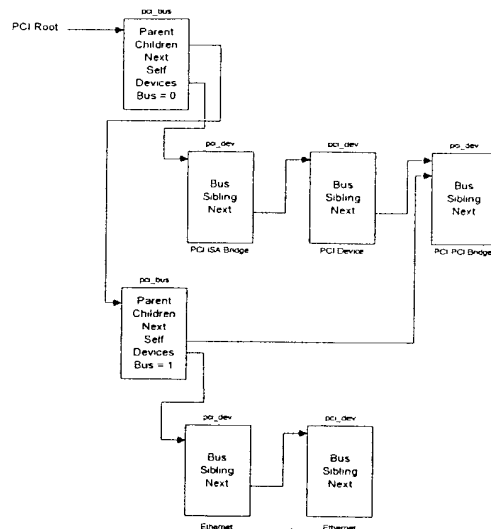
The PCI initialisation code in Linux is broken into three logical parts:

1. PCI Device Driver
  - a. This pseudo-device driver searches the PCI system starting at Bus 0 and locates all PCI devices and bridges in the system. It builds a linked list of data structures describing the topology of the system. Additionally, it numbers all of the bridges that it finds.
2. PCI BIOS
  - a. This software layer provides the services described in bib-pci-bios-specification Even if there is no BIOS, there is equivalent code in the Linux kernel providing the same functions.
3. PCI Fix-up
  - a. System specific fix-up code tidies up the system specific loose ends of PCI initialization.

### 19.4 The Linux Kernel PCI Data Structures

A-109

## Little Joe Functional Specification



As the Linux kernel initializes the PCI system it builds data structures mirroring the real PCI topology of the system. The figure above shows the relationships of the data structures that it would build for the example PCI system described at the beginning of this appendix.

Each PCI device (including the PCI-PCI Bridges) is described by a `pci_dev` data structure. Each PCI bus is described by a `pci_bus` data structure. The result is a tree structure of PCI buses each of which has a number of child PCI devices attached to it. As a PCI bus can only be reached using a PCI-PCI Bridge (except the primary PCI bus, bus 0), each `pci_bus` contains a pointer to the PCI device (the PCI-PCI Bridge) that it is accessed through. That PCI device is a child of the the PCI Bus's parent PCI bus.

Not shown in the above is a pointer to all of the PCI devices in the system, `pci_devices`. All of the PCI devices in the system have their `pci_dev` data structures queued onto this queue. This queue is used by the Linux kernel to quickly find all of the PCI devices in the system.

#### 19.4 The PCI Device Driver

The PCI device driver is not really a device driver at all but a function of the operating system called at system initialization time. The PCI initialization code must scan all of the PCI buses in the system looking for all PCI devices in the system (including PCI-PCI bridge devices).

It uses the PCI BIOS code to find out if every possible slot in the current PCI bus that it is scanning is occupied. If the PCI slot is occupied, it builds a `pci_dev` data structure describing the device and links into the list of known PCI devices (pointed at by `pci_devices`).

The PCI initialization code starts by scanning PCI Bus 0. It tries to read the Vendor Identification and Device Identification fields for every possible PCI device in every possible PCI slot. When it finds an occupied slot it builds a `pci_dev` data structure describing the device.

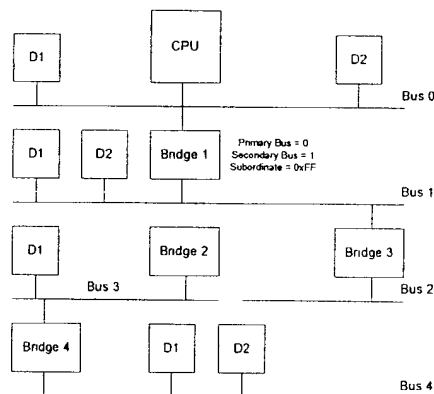
A-105

All of the `pci_dev` data structures built by the PCI initialization code (including all of the PCI-PCI Bridges) are linked into a singly linked list; `pci_devices`.

If the PCI device that was found was a PCI-PCI bridge then a `pci_bus` data structure is built and linked into the tree of `pci_bus` and `pci_dev` data structures pointed at by `pci_root`. The PCI initialization code can tell if the PCI device is a PCI-PCI Bridge because it has a class code of 0x060400. The Linux kernel then configures the PCI bus on the other (downstream) side of the PCI-PCI Bridge that it has just found. If more PCI-PCI Bridges are found then these are also configured. This process is known as a depth-wise algorithm; the system's PCI topology is fully mapped depth-wise before searching breadthwise. Looking at reference model at the beginning of the appendix, Linux would configure PCI Bus 1 with its Ethernet and SCSI device before it configured the video device on PCI Bus 0.

As Linux searches for downstream PCI buses it must also configure the intervening PCI-PCI bridges' secondary and subordinate bus numbers. This is described in detail in Section `pci-pci-bus-numbering` below.

### 19. *Configuring PCI-PCI Bridges - Assigning PCI Bus Numbers*



For PCI-PCI bridges to pass PCI I/O, PCI Memory or PCI Configuration address space reads and writes across them, they need to know the following:

1. Primary Bus Number
  - a. The bus number immediately upstream of the PCI-PCI Bridge.
2. Secondary Bus Number
  - a. The bus number immediately downstream of the PCI-PCI Bridge.
3. Subordinate Bus Number
  - a. The highest bus number of all of the buses that can be reached downstream of the bridge.

**A-106**

Case 2:23-cv-00202-JRG-RSP Document 86-5 Filed 08/12/24 Page 116 of 156 PageID #: 2550

## Little Joe Functional Specification

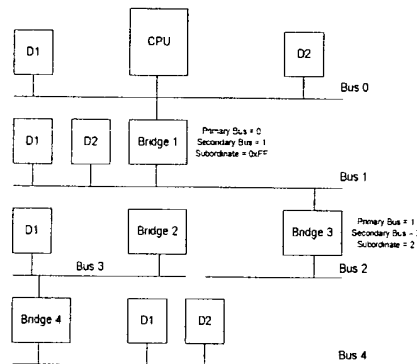
**b. PCI I/O and PCI Memory Windows**

The window base and size for PCI I/O address space and PCI Memory address space for all addresses downstream of the PCI-PCI Bridge.

The problem is that at the time when you wish to configure any given PCI-PCI bridge you do not know the subordinate bus number for that bridge. You do not know if there are further PCI-PCI bridges downstream and if you did, you do not know what numbers will be assigned to them. The answer is to use a depth-wise recursive algorithm and scan each bus for any PCI-PCI bridges assigning them numbers as they are found. As each PCI-PCI bridge is found and its secondary bus numbered, assign it a temporary subordinate number of 0xFF and scan and assign numbers to all PCI-PCI bridges downstream of it. This all seems complicated but the worked example below makes this process clearer.

**a. PCI-PCI Bridge Numbering: Step 1**

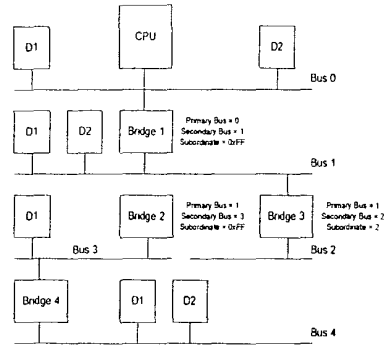
Taking the topology in under “Assigning PCI Bus Numbers”, the first bridge the scan would find is Bridge1. The PCI bus downstream of Bridge1 would be numbered as 1 and Bridge1 assigned a secondary bus number of 1 and a temporary subordinate bus number of 0xFF. This means that all Type 1 PCI Configuration addresses specifying a PCI bus number of 1 or higher would be passed across Bridge1 and onto PCI Bus 1. They would be translated into Type 0 Configuration cycles if they have a bus number of 1 but left un-translated for all other bus numbers. This is exactly what the Linux PCI initialization code needs to do in order to go and scan PCI Bus 1.

**b. PCI-PCI Bridge Numbering: Step 2**

Linux uses a depth-wise algorithm and so the initialization code goes on to scan PCI Bus 1. Here it finds PCI-PCI Bridge2. There are no further PCI-PCI bridges beyond PCI-PCI Bridge2, so it is assigned a subordinate bus number of 2 which matches the number assigned to its secondary interface. The diagram above shows how the buses and PCI-PCI bridges are numbered at this point.

A-107

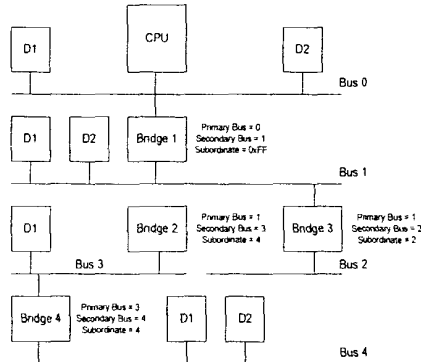
00000000000000000000000000000000



### c. PCI-PCI Bridge Numbering: Step 3

The PCI initialization code returns to scanning PCI Bus 1 and finds another PCI-PCI bridge, Bridge3. It is assigned 1 as its primary bus interface number, 3 as its secondary bus interface number and 0xFF as its subordinate bus number.

Type 1 PCI configuration cycles with a bus number of 1, 2 or 3 will be correctly delivered to the appropriate PCI buses.



1.47

### d. PCI-PCI Bridge Numbering: Step 4

Linux starts scanning PCI Bus 3, downstream of PCI-PCI Bridge3. PCI Bus 3 has another PCI-PCI bridge (Bridge4) on it, it is assigned 3 as its primary bus number and 4 as its secondary bus number. It is the last bridge on this branch and so it is assigned a subordinate bus interface number of 4. The initialization code returns to PCI-PCI Bridge3 and assigns it a subordinate bus number of 4. Finally, the PCI initialization code can assign 4 as the subordinate bus number for PCI-PCI Bridge1.

A-108

Little Joe Functional Specification

**e. PCI BIOS Functions**

The PCI BIOS functions are a series of standard routines which are common across all platforms. For example, they are the same for both Intel and Alpha AXP based systems. They allow the CPU controlled access to all of the PCI address spaces. Only Linux kernel code and device drivers may use them.

**1.48**

**f. PCI Fixup**

For PowerPC based systems without a BIOS to set up PCI configuration needs to happen to:

1. Allocate PCI I/O and PCI Memory space to each device.
2. Configure the PCI I/O and PCI Memory address windows for each PCI-PCI bridge in the system
3. Generate Interrupt Line values for the devices; these control interrupt handling for the device.

The next subsections describe how that code works.

**1.49**

**g. Finding Out How Much PCI I/O and PCI Memory Space a Device Needs**

Each PCI device found is queried to find out how much PCI I/O and PCI Memory address space it requires. To do this, each Base Address Register has all 1's written to it and then read. The device will return 0's in the don't-care address bits, effectively specifying the address space required.

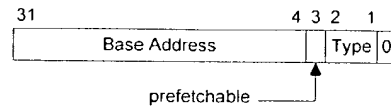
A-109

08/12/2024 11:00:07

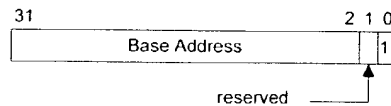
## 1.49.1

**a. PCI Configuration Header: Base Address Registers**

Base Address for PCI Memory Space



Base Address for PCI I/O Space



There are two basic types of Base Address Register, the first indicates within which address space the devices registers must reside; either PCI I/O or PCI Memory space. This is indicated by Bit 0 of the register. The diagrams above show the two forms of the Base Address Register for PCI Memory and for PCI I/O.

To find out just how much of each address space a given Base Address Register is requesting, you write all 1s into the register and then read it back. The device will specify zeros in the don't-care address bits, effectively specifying the address space required. This design implies that all address spaces used are a power of two and are naturally aligned.

For example when you initialize the DECChip 21142 PCI Fast Ethernet device, it tells you that it needs 0x100 bytes of space of either PCI I/O or PCI Memory. The initialization code allocates it space. The moment that it allocates space, the 21142's control and status registers can be seen at those addresses.

**b. Allocating PCI I/O and PCI Memory to PCI-PCI Bridges and Devices**

Like all memory the PCI I/O and PCI memory spaces are finite, and to some extent scarce. The PCI Fixup code for PowerPC systems has to allocate each device the amount of memory that it is requesting in an efficient manner. Both PCI I/O and PCI Memory must be allocated to a device in a naturally aligned way. For example, if a device asks for 0xB0 of PCI I/O space then it must be aligned on an address that is a multiple of 0xB0. In addition to this, the PCI I/O and PCI Memory bases for any given bridge must be aligned on 4K and on 1Mbyte boundaries respectively. Given that the address spaces for downstream devices must lie within all of the

A-110

## Little Joe Functional Specification

upstream PCI-PCI Bridge's memory ranges for any given device, it is a somewhat difficult problem to allocate space efficiently.

The algorithm that Linux uses relies on each device described by the bus/device tree built by the PCI Device Driver being allocated address space in ascending PCI I/O memory order. Again a recursive algorithm is used to walk the `pci_bus` and `pci_dev` data structures built by the PCI initialization code. Starting at the root PCI bus (pointed at by `pci_root`) the BIOS fixup code:

1. Aligns the current global PCI I/O and Memory bases on 4K and 1 Mbyte boundaries respectively.
2. For every device on the current bus (in ascending PCI I/O memory needs)
  - i. Allocates it space in PCI I/O and/or PCI Memory
  - ii. Moves on the global PCI I/O and Memory bases by the appropriate amounts
  - iii. Enables the device's use of PCI I/O and PCI Memory
3. Allocates space recursively to all of the buses downstream of the current bus. Note that this will change the global PCI I/O and Memory bases.
4. Aligns the current global PCI I/O and Memory bases on 4K and 1 Mbyte boundaries respectively and in doing so figure out the size and base of PCI I/O and PCI Memory windows required by the current PCI-PCI bridge.
5. Programs the PCI-PCI bridge that links to this bus with its PCI I/O and PCI Memory bases and limits.
6. Turns on bridging of PCI I/O and PCI Memory accesses in the PCI-PCI Bridge. This means that if any PCI I/O or PCI Memory addresses seen on the Bridge's primary PCI bus that are within its PCI I/O and PCI Memory address windows will be bridged onto its secondary PCI bus.

Taking the PCI system in at the beginning of this appendix as our example the PCI Fixup code would set up the system in the following way:

1. Align the PCI bases
  - a. PCI I/O is 0x4000 and PCI Memory is 0x100000. This allows the PCI-ISA bridges to translate all addresses below these into ISA address cycles
2. The Video Device
  - a. This is asking for 0x200000 of PCI Memory and so we allocate it that amount starting at the current PCI Memory base of 0x200000 as it has to be naturally aligned to the size requested. The PCI Memory base is moved to 0x400000 and the PCI I/O base remains at 0x4000.
3. The PCI-PCI Bridge
  - a. We now cross the PCI-PCI Bridge and allocate PCI memory there, note that we do not need to align the bases as they are already correctly aligned:
    - i. The Ethernet Device
      1. This is asking for 0xB0 bytes of both PCI I/O and PCI Memory space. It gets allocated PCI I/O at 0x4000 and PCI Memory at 0x4000B0. The PCI Memory base is moved to 0x4000B0 and the PCI I/O base to 0x40B0.

A-III





[13 pages]

60423696 1120407

EV188390615

B

## MULTI MAC CONTROL DESIGN DOCUMENT VER1.4

The Little Joe system consists of co-located APs connected to a Butler Matrix. Transmitting a downlink packet on one AP completely destroys any uplink packets being received simultaneously on another AP. Allowing each AP to operate independently causes serious performance issues [1]. The Multi MAC Controller (MMC) controls the transmissions of each AP to prevent data suicide. The rationale behind the MMC, its requirements and various solutions are discussed in [2].

The approach is to choose an initial solution that is simple, with basic hooks to manage the impact of overlapping subnets, and be robust under all failure conditions. Ensure that all the lines required for the implementation of various features are made available to the mmc hardware block, and that the hardware device (CPLD, FPGA) is big enough to support the entire functionality.

The optional features need not be implemented initially, but they may need to be implemented later if the MMC software needs additional control.

### OVERVIEW

The MMC consists of hardware logic on an FPGA/PLD and control software on the host.

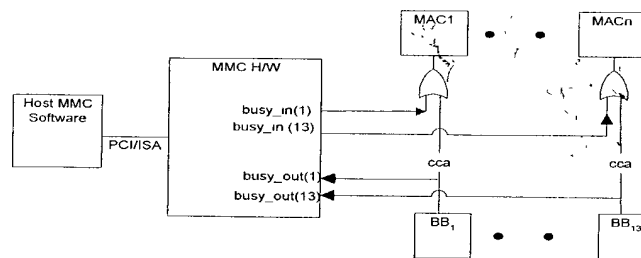


Figure 1 MMC Overview

The CCA busy line is always propagated to the MAC from its own beam (CCA busy lines are sniffed, not spoofed). If carrier sense (CS) is available then the MMC uses only the CS and not energy detect (ED), as the ED line is triggered even in the presence of non correlated interference, the objective here is to protect only valid packets.

In the Agere chipset the MBUSY line indicates CS. In the prism 2.0 chipset cca line indicates cs/ed based on the configured mode.

B-1

Prism 2.5 chipset does not expose the cca line, and therefore mmc would require an external cca detector that provides the input to MMC and a cca generator which is triggered by the cca\_cnt lines of the MMC to inject a carrier signal with valid preamble to radio causing the BB to assert CCA line to the MAC.

The input to the MMC labeled as busy\_out is active high when the channel is sensed to be busy. The output is labeled as busy\_in, is active high when the MMC indicates to the MAC that the channel is busy.

B-2

600423606 117402

# MMC HARDWARE

The MMC hardware selectively ORs together the 13 BUSY output lines from the BB on a per channel basis, to generate 13 BUSY input lines for the MACs. A global timer limiter (GTL) function is used per channel to identify failure conditions. The hardware functionality is shown in Fig2.

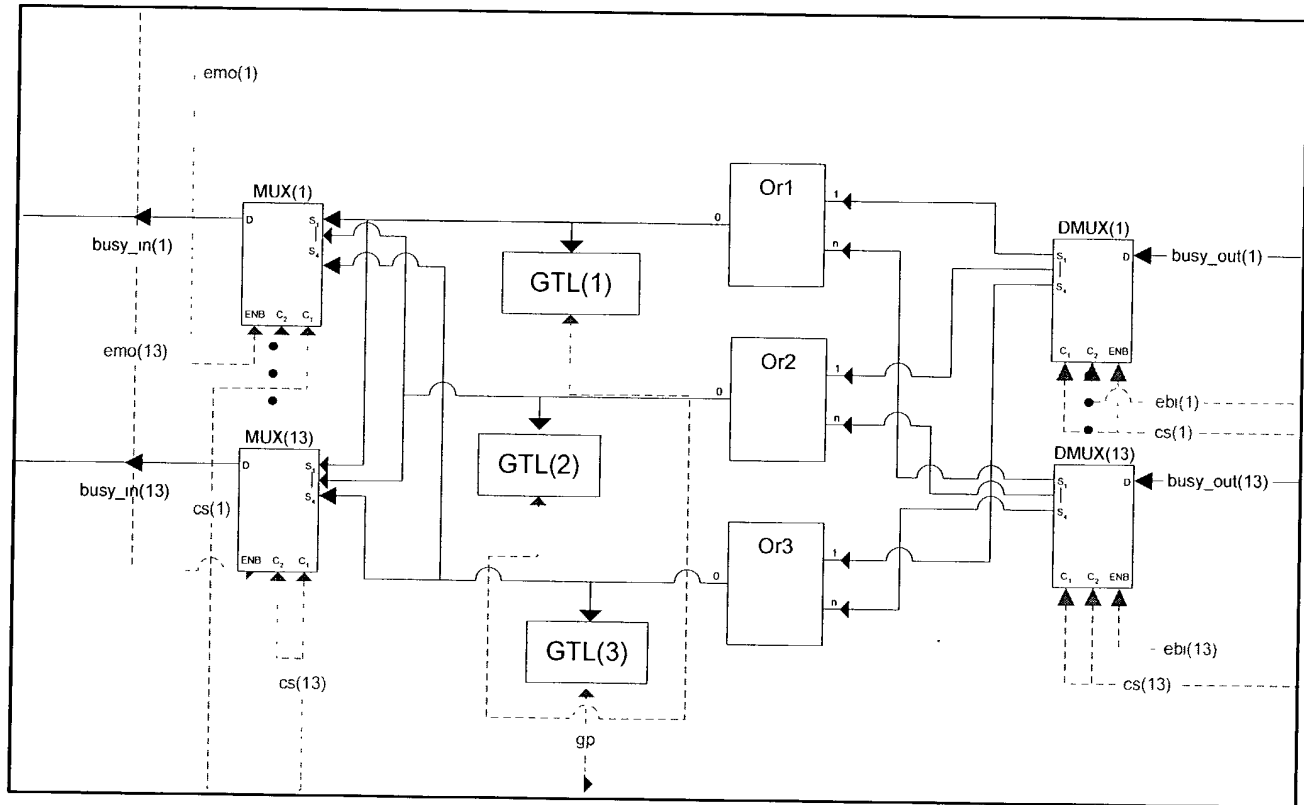


Figure 2 Hardware Functional Block Diagram

B-3



Case 2:23-cv-00202-JRG-RSP Document 86-5 Filed 08/12/24 Page 126 of 156 PageID #: 2560

INPUT BUSY DMUX:

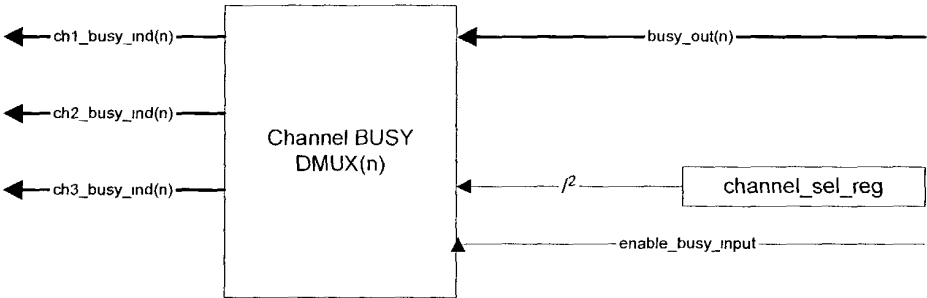


Figure 3 Input BUSY DMUX

- 1. Each busy\_out input to MMC may be turned on/off by the control software using enable\_busy\_out bit of the aMMCCControlRegister (2bit). Note: In the future hardware functional blocks may access this.
- 2. At reset all t/p lines are turned off.

Table 2 BUSY DMUX Truth Table

enable_busy_out	1	1	1	1	x	0
channel_sel_reg(0)	0	1	0	1	x	x
channel_sel_reg(1)	0	0	1	1	x	x
busy_out	1	1	1	x	0	x
ch1_busy_ind	1	0	0	0	0	0
ch2_busy_ind	0	1	0	0	0	0
ch3_busy_ind	0	0	1	0	0	0

Case 2:23-cv-00202-JRG-RSP Document 86-5 Filed 08/12/24 Page 127 of 156 PageID #: 2561

OUTPUT BUSY PROPAGATION MUX:

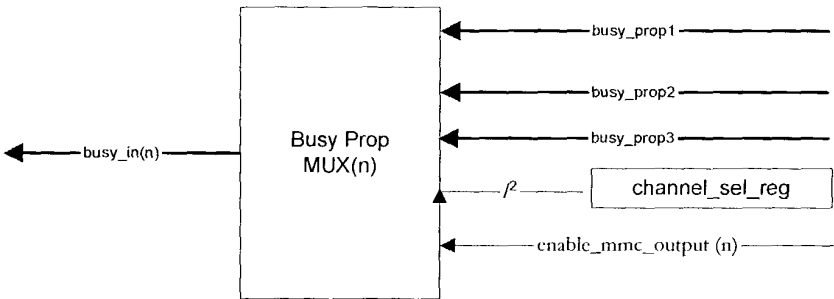


Figure 4 Busy Propagation MUX

- The propagation of the output of MMC to each MAC is controlled by the software using the enable\_mmc\_cca\_output bit of the aMMCCControlRegister (2bit).
- At reset the o/p line is turned off.

Table 3 CCA Propagation Mux Truth Table

enable_mmc_output	1	1	1	1	x	0
channel_sel_reg(0)	0	1	0	1	x	x
channel_sel_reg(1)	0	0	1	1	x	x
busy_prop1	1	0	0	x	0	x
busy_prop2	0	1	0	x	0	x
busy_prop3	0	0	1	x	0	x
busy_in	1	1	1	0	0	0

B-6

Case 3:23-cv-00202-JRG-RSP Document 86-5 Filed 08/12/24 Page 128 of 156 PageID #: 2562

## GLOBAL TIME LIMITER

A Global Time Limiter (GTL) assists the MMC to restrict the MMC output line from being held up active for more than a period defined by aMMCWatchDogTimerLimit (gtl\_timer\_register, 32 bit, units of kus). The same value is used for all channels.

- Start the GTL timer on rising edge of the MMC output.
- The timer is stopped and reloaded with aMMCWatchDogTimerLimit on the falling edge.
- If this timer expires, indicate to the MMC software via interrupt, the software is expected to stop operation of MMC, and diagnose the condition.

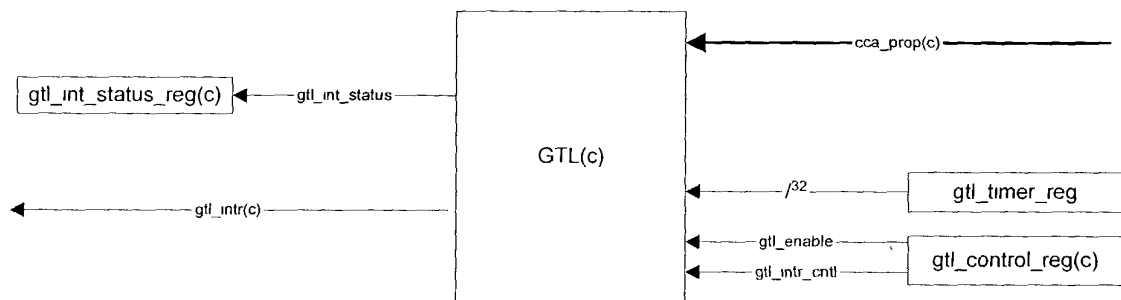


Figure 5 GTL Block Diagram

B-7



08/12/24 09:55:11 AM

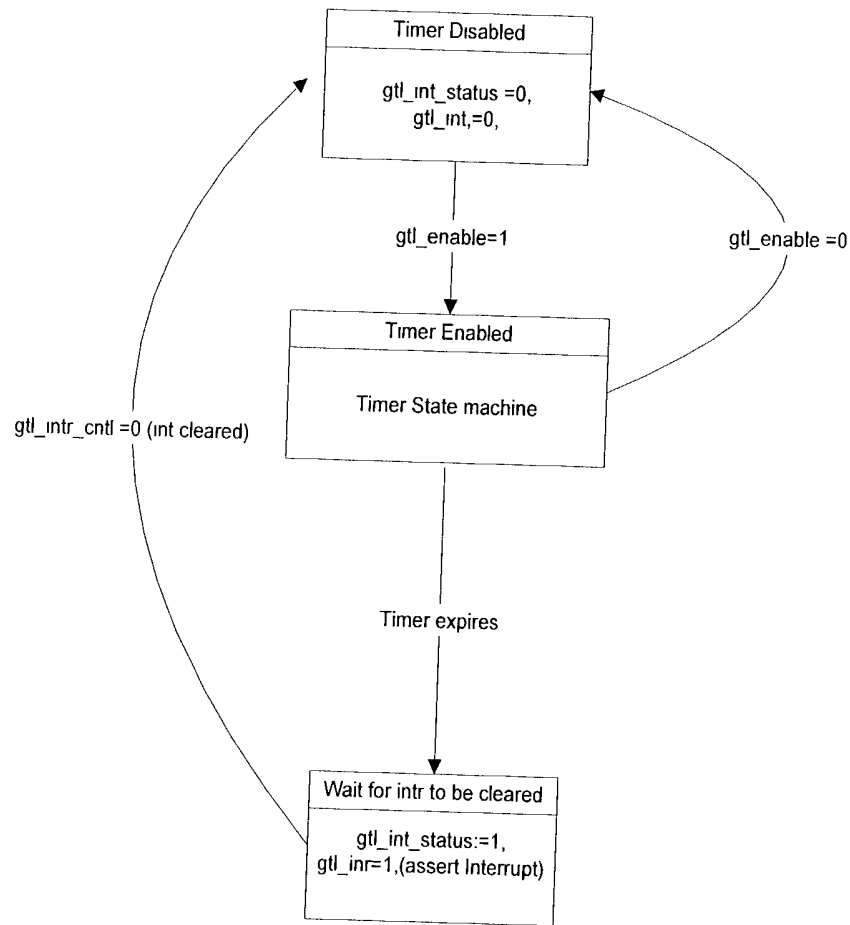
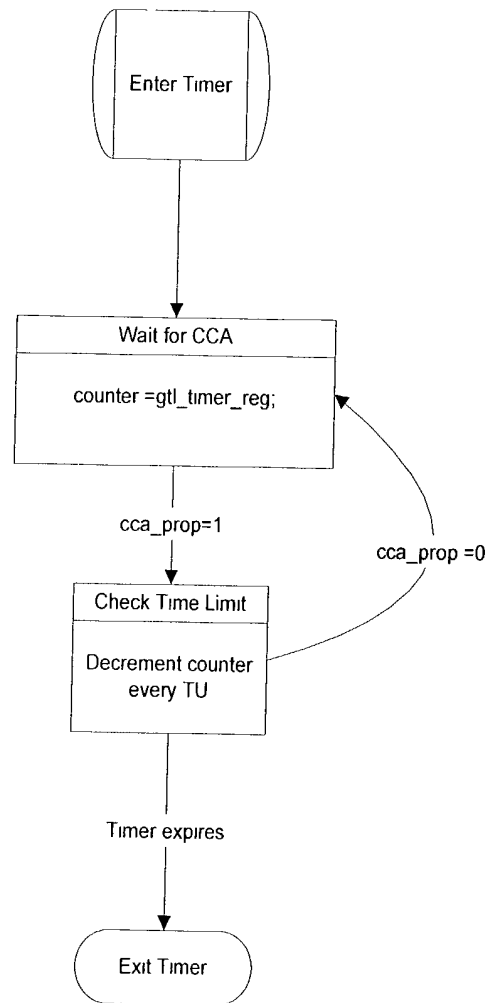


Figure 6GTL State Machine

B8



### Figure 7GTL Timer State Machine

60423696, 3, 704012

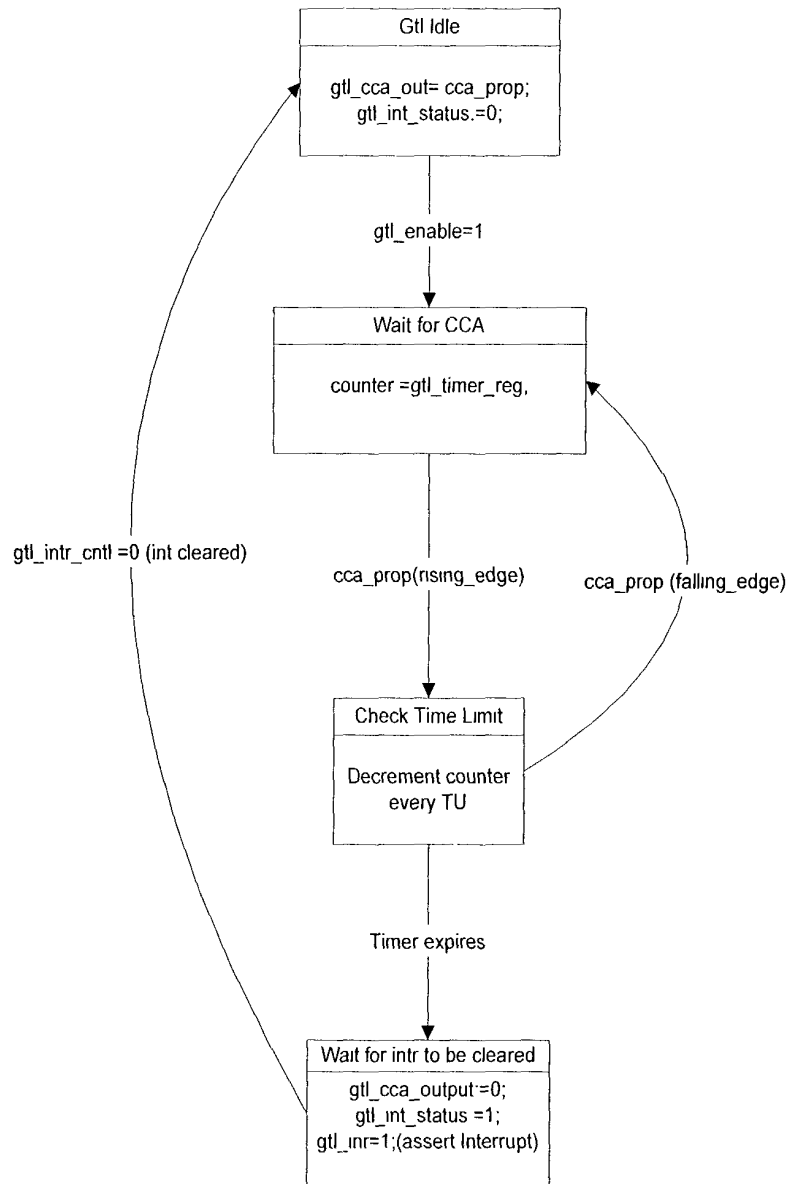


Figure 8 Old GTL State Machine (TBdeleted)

One way to turn off the whole MMC would be to individually turning off each dmus/mux.

B-10

CONFIDENTIAL

#### MMC SOFTWARE

- MMC is on a per channel basis. The software configures the multiplexers based on the current channel allocation by programming the aChannelSelectionRegister for each radio. This ensures that only BUSYs from one channel are used for the MMC function.
- Software uses the aMMCControlRegister to selectively exclude the BUSY line for a beam from being considered by MMC when:
  - There are no associated clients on that beam.
  - The observed overlapping BSS load on that beam exceeds a certain level aProportionPerBeamOBSSLoadHigh [REDACTED] compared to uplink BSS load. OBSS load is estimated by software using the scanning radio operating in promiscuous mode. Note: aProportionPerBeamOBSSLoadHigh is ideally only load from an overlapping ESS.
- BUSYlines that are excluded will need to be included when the situation changes.
  - New clients associate on a beam.
  - The observed OBSS loads compared to uplink load falls below a threshold aProportionPerBeamOBSSLoadLow [REDACTED].
- Program the GTL to an appropriate value aMMCWatchDogTimerLimit [REDACTED].
- Handle the GTL interrupt: The intention of GTL is to detect failure conditions. On failure the MMC software will need to identify the cause, and if required reset the h/w blocks associated with the BUSY line that is stuck.
  - One way of doing this will to selectively exclude each line and wait for a multiple of the aMMCWatchDogTimerLimit duration and see if the timer expires.
  - It is not expected that this operation will not disrupt the system, the system may move beyond the operating point until the MMC is restored or the load control algorithms react.
- The software is able to turn off the MMC output from being delivered to each MAC. This feature may be used in the future if downlink in one direction is being starved by heavy uplink in another direction.

B-11

CONFIDENTIAL

#### APPENDIX

The sections below discuss other features that were discussed. It was agreed that these features shall not be mandatory however they may be implemented later if required.

#### CHANNEL UTILIZATION MEASUREMENT BLOCK (CUMB)

This functionality is optional, and would be useful for the MMC software to re-enable BUSY input to the MMC, and trouble shoot when the GTL expires.

- Calculate % time each beam is busy, and used to choose directions for including the BUSY from that beam for MMC. Measure the total duration for which the BUSY is high within a time period of aCUMeasurementDuration ( $2^x$ ) kus. This value is then averaged over aCUMeasurementPeriod ( $2^x$ ) measurements.
- In addition to the measure based on CS, ED may be measured separately to detect the presence of non correlated interference. This information may be provided to dynamic channel allocation mechanism.
- There may be need to count overlapping BSS frames separately.

#### MMC Software :

The Channel utilization measure will now be used to handle the GTL interrupt. Exclude lines that have utilization exceeding some value [REDACTED]. Monitor the values over time and restore lines if needed by resetting h/w.

#### INPUT LIMITER TIMERS

The following timers should be supported (optional feature).

1. i/p limiting timer : started on the rising edge of BUSY that does not allow each BUSY input to MMC to remain high for more than a duration that is programmed by the host. For eg. this could be set to max packet length at a particular rate to limit the max duration of a packet that can avail of the MMC support to avoid data suicide. This timer will be adaptively controlled using an appropriate algorithm on host, considering xxx factors. However being unfair to users located beyond a particular range may not be appropriate.
2. i/p off timer: a timer that is started once the i/p-limiting-timer is triggered. Until this timer expires that BUSY line would be ignored for MMC.

B-12

#### GOOD PACKET FILTER (GPF)

The GPF is optional and the goal is to turn off the 1/p switch when the BUSY is asserted by a packet we are not interested in protecting, this is for a duration that is equal to the expected packet length.

- Sniff RX lines, and compare Address1 field of the received frame, if there is no match then pull down the input to MMC for the duration of that frame and its response. (address overlapping subnets, useful for big frames).
- Avoid packets being received that are already corrupted by downlink transmissions. If BUSY was high when there was a down link transmission (need to spoof txlines, or look at address2), then wait for the next rising edge before including BUSY again.
- Two overlapping uplink packets on the same beam just choose the first one if its a good packet, and exclude the remainder of the other packet. (partly covered by above case, but can avoid preamble this way).
- if the PLCP CRC is bad, but CS indicates high, then pull down input to MMC.
- Consider RSSI on the beam to decide if BUSY should be included into MMC (need more investigation with agere chipset).

#### Programming support for GPF

- Address 1 Programmability
- Turn on and off each feature

#### Load/ Fairness Based Control

1. Selectively including directions when the load is high in one direction and not in other direction, based on downlink load, and network policy.
2. Increased lower PHY rate packets are observed in one direction, again based on policy.
3. The channel utilization measure may be used for this purpose.

60923695-130402

EW188390615

C

[11 pages]

Little Joe Multi-MAC Controller

( aka CCA Glue Logic)

C-1

# Objectives

- Avoid Data Suicide
  - Prevent downlink transmissions when there are good uplink transmissions on the same channel.
- Out of the scope
  - Measure Channel Utilization
    - Provide a measure of channel use per beam for traffic load control,.
    - Percentage of time channel is busy(CS) with valid uplink packets.
    - Load Control Team to investigate if the host based solution is sufficient.
  - Use of CCA for the inter-mod problem.
  - Measuring ED busy to detect non correlated interferer for dynamic channel allocation, better done by scanning radio.
  - Managing ACK Suicide.
  - CCA input to MAC from same BB turned off based on RSSI (interference handling).
  - Vivato MAC/BBP chips.
  - Stopping beacons when there are no associated STA.

C-2



# Assumptions

- Co-located APs connected to a Butler Matrix.
- Each AP can sense 50% of the STAs in the 11Mbps coverage area.
  - Rx windowing significantly reduces the number of STAs that an AP can sense.
- Transmitting a downlink packet completely destroys any uplink packets being received simultaneously.
- Each AP receive down link transmissions from other APs, unless there is a local collision, and assert NAV based on the duration field.
- Downlink transmissions are sensed by 90% of all STA in the 11Mbps Coverage area if CBF is used.

C-3

# Do we need MMC?

- CBF is easy and cheap to implement, and “will” be implemented if FCC permits.
- The question is how does CBF with CCA compare to just CBF?.
- Simulation (CCA-OR) Results Indicate – Yes.
  - *Downlink Load fixed at 1.3 Mb/s, CCA - 27 %, CBF- 17 %, CCA+ CBF- 42 %*
  - *Downlink Load fixed at 650 Kb/s, CCA - 32 %, CBF - 17 %, CCA+ CBF- 49 %*
  - MMC helps improve the operating point significantly in certain load scenarios.
- Discussion
  - Simulation assumes that each AP can sense none of the STAs associated with other APs, in reality if each AP can sense 50% of the STAs in the 11Mbps coverage area. Improvement is 1/2 what is indicated.
  - The results in the report show best case CBF, i.e, all STA listen to CBF. Other simulations indicate that with fewer STAs covered by CBF linear degradation is observed. So CBF is 10% worse than indicated.
- Conclusion:
  - We need MMC.
  - if the 50% assumption is correct then its not as critical.
  - If FCC does not permit CBF then MMC is important.
  - MMC should not be the gating factor for the product.

C-9

# Simple CCA-OR ?

- How bad would simple CCA-OR solution perform with OBSS,(non correlated interference, may be avoided by using cs not ed) in selected directions?.
- Ideally you do not want to prevent backoff count down in one direction when the channel is busy in another direction. Just want to prevent actual transmissions that destroy valid packets.
  - Consider the case where you have many STA in one beam, keeping the beam busy.
  - Consider the case where some of the clients in one beam are using a much lower phy rate.
  - Don't want to starve downlink completely because of directional failure.
- [REDACTED]
- [REDACTED]

C-5

# MMC Hardware

One

Idea: Choose a simple solution with basic hooks to manage impact of overlapping subnets, and be robust under all failure conditions.

- CCA always propagated to the MAC from its own beam, i.e., CCA lines sniffed, not spoofed.
- Consider only CS not ED as we are interested in protecting only valid packets.
- Each CCA is input to MMC through a switch.
- The output of MMC propagated to each MAC through a switch.

C-6

## MMC Hardware contd.

- Global Time Limiter
  - Start a timer when MMC output goes high so that the MMC output shall not remain active for more than aMMCWatchDogTimer.
  - If this timer expires, stop and indicate to the host software.

C-7

# MMC Software

- MMC on a per channel basis, Only propagate CCAs for Radios on the same channel
  - Choose groups of radios, programmable by software that does channel allocation.
- Software selectively excludes some of CCA lines from being considered by MMC
  - When there are no associated clients on the beam.
  - When the observed load on overlapping BSS exceeds certain level [REDACTED]. OBSS load estimated by software using scanning radio operating in promiscuous mode .
  - When the load is high in one direction and not in other direction?
  - When increased lower PHY rate packets are observed in one direction?
- Able to turn off propagation of MMC output.

C-8

# Consider only good uplink transmissions

Good to have:

- Sniff rx, and compare Address1 field, if there is no match then pull down from input to MMC, (address overlapping subnets, useful for big frames)
- Avoid packets being received that are already corrupted by downlink transmissions.
- Two overlapping uplink packets just choose the first if its a good packet, and exclude the remainder of the other packet.
- if the PLCP CRC is bad, but CS indicates high, then pull down input to MMC.
- Consider RSSI on the beam to decide if CCA should be included into MMC (need more investigation with agere chipset.).

C-9

# Other Features

- Timer started on the rising edge of CCA that does not allow each CCA input to MMC to remain high for more than a duration, programmed by the host. For eg. this could be set to max packet length at a particular rate to limit the max duration of a packet that can avail of the MMC support to avoid data suicide.
- Adaptively controlling the above timer value using an appropriate algorithm on host, considering xxx factors, and another timer that controls when CCA would be considered again.
- Length field extension (11b timer), check if agere does it, else do it.
- Exceed any cca busy (tx) end by a delta for expected response (rare corner case ignore)
- Calculate % time each beam is busy, and used to choose directions for including the CCA from that beam for MMC.

C-10



# Utilization Measure requirements

- Provide a measure of channel use per beam for traffic load control.
  - The basic idea is to provide a measure of the wasted uplink transmissions, and thereby quantify the impact of hidden beam problem.
- The solution would also consider retry bits, sequence numbers, number of rx packets received, with CRC errors, run on the host, (need more investigation).
- Percentage of time channel is busy with valid uplink packets.
- Do not include transmissions in OBSS, or count them separately.
- Do not include transmissions by other co-located radios.
- [REDACTED]
- [REDACTED]

611

CONFIDENTIAL

EW188390615

D

[12 pages]

MAC problems: Using multiple  
co-located APs with smart  
antennas

Little Joe 1 Context ver 2.0



D-1

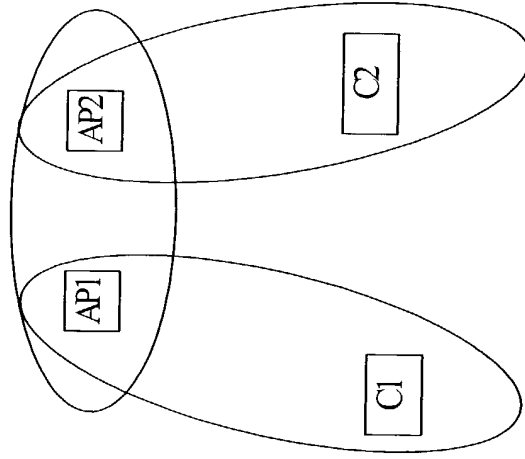
# Background

- Co-located APs connected to a Butler Matrix.
- Each AP can sense 50% of the STAs in the 11Mbps coverage area.
  - Rx windowing significantly reduces the number of STAs that an AP can sense.
- Transmitting a downlink packet completely destroys any uplink packets being received simultaneously.
- Each AP receive down link transmissions from other APs, unless there is a local collision, and assert NAV based on the duration field.
- Downlink transmissions are sensed by 90% of all STA in the 11Mbps Coverage area if CBF is used.

0-2

# Example Scenario

- Consider two co-located APs AP1, AP2, and two STAs C1 and C2.
- Develop a Scenario Matrix, each entry representing the scenario when transmission from the device on the column starts at least aSlotTime after transmissions from the device on the row.
  - Ignore 802.11 collisions.



D-3

# Hidden Terminal Problem

- Downlink CTS is not heard by STAs outside the coverage of the beam.
- Uplink data frames in one beam are not heard by STA in other directions.
- Because of large range, the uplink data frames on one beam are not heard by all the STA associated with the AP on that beam.
- Large number of hidden terminals than normal APs.
- Result: Uplink transmission while another uplink is in progress.

**D-4**

# Ack Suicide

- ACK responses to an uplink packet collide with other uplink packets.

## Note:

- ACK transmissions do not look at virtual carrier sense or physical carrier sense.
- But CTS transmissions happen only if channel is available (CCA, NAV=0).

DS

# Data Suicide

- Downlink frames other than ACK colliding with Uplink Frames.
- Includes Data Packets, CTS, and other responses that are not transmitted if NAV/CTS is asserted.
- Problem arises because of multiple MACs operating independently.

D-6

# Hidden Beam Problem

- Uplink transmissions are not aware of downlink transmissions in other directions.
- Uplink frames collide with an ongoing down link frame.
  - Including downlink ack, data, control etc..


D-7



60423696, 1.10402

D-8

# Problem Matrix

 Row tx before column	A1	A2	C1	C2
A1	x	x	x	Hidden Beam
A2	x	x	Hidden Beam	x
C1	x	Data Suicide	x	Ack Suicide
C2	Data Suicide	x	Ack Suicide	x

# Downlink Local Collisions

- Down link transmission from two or more co-located APs collide, only one packet is transmitted by the switch.
- Low probability of this happening.

D-9

# Auto Rate Fall Back

- Auto rate fall back algorithms typically work on the basis of counting ACK losses or packets drops after exceeding retry limit. ARF restoration algorithms are often less aggressive.
- Large number of uplink data packets are corrupted because of the problems described, i.e ACKs not transmitted.
- Depending on traffic load, several uplink packets are dropped after retry.
- STAs trigger auto rate fall back and may not restore rate.
- Might need to limit operation to 11Mbps.
  - 11 Mbps as Basic Rate Set, limit to 11Mbps coverage.
  - Limiting to Highest rate for 11g is not an option.

D-10

# Big Packet Small Packet Unfairness

- Because of ACK suicide, large uplink frames have smaller probability of success.
- STA with smaller average packet size gets higher uplink throughput.
- May [REDACTED] manage ACK transmissions.

0-11